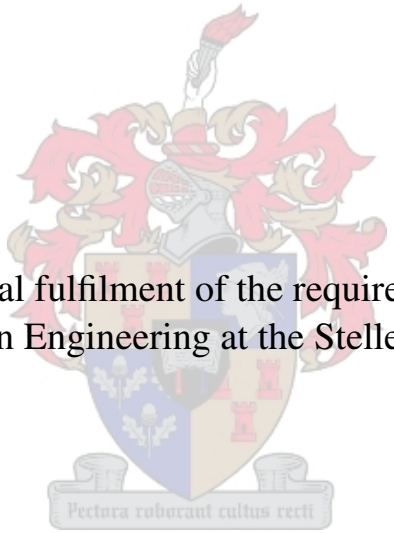
Energy Storage in Composite Flywheel Rotors

Petrus J. Janse van Rensburg



Thesis presented in partial fulfilment of the requirements for the degree of
Master of Science in Engineering at the Stellenbosch University

Faculty of Engineering
Department of Mechanical and Mechatronic Engineering

Supervisor: Prof. Albert A. Groenwold
December 2011

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the owner of the copyright thereof (unless to the extent explicitly otherwise stated) and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: .....

Abstract

As the push continues for increased use of renewables on the electricity grid, the problem of energy storage is becoming more urgent than ever. Flywheels with wound, composite rotors represent an efficient and environmentally friendly option for energy storage. They have already been applied successfully for voltage control on electrical rail networks and for bridging power in backup UPS systems, but lately they have also proven useful for grid-scale frequency regulation.

For flywheels to be deployed on a wider scale, the high cost associated with the technology will have to be addressed. An important driver of cost is the density at which energy can be stored. Currently, flywheel designs do not consistently achieve high energy density, and this study investigates the reasons for this.

A critical analysis is made of the design methodologies that have been proposed in the available literature, and some improvements are suggested. Most notably it is shown that significant improvements in energy density may be possible if the design optimization problem is formulated carefully.

In addition, the problem of material selection is discussed, because material properties have a significant influence on energy density. Some guidance is given for flywheel designers on how to choose an optimal set of materials without invoking undue computational effort. It is hoped that these suggestions may be carried forward as a topic of further research.

Keywords: Renewable energy, energy storage, flywheels, flywheel rotors, energy density

Opsomming

Namate die aanvraag vir hernubare energie op die elektrisiteit netwerk vergroot, word die probleem van energie berging van kardinale belang. Vliegwiele met silindriese rotors van samegestelde materiale bied 'n effektiewe en omgewingsvriendelike opsie vir energieberging. Hierdie tipe vliegwiele is reeds suksesvol aangewend vir spanningsbeheer op elektriese spoornetwerke en om oorbruggingskrag te voorsien aan rugsteun sisteme. Meer onlangs is hulle ook nuttig bewys vir die regulasie van frekwensie op die elektrisiteit netwerk.

Grootskaalse aanwending van vliegwiele kan egter slegs oorweeg word indien die hoë koste van die tegnologie aangespreek word. Een van die onderliggende redes vir die hoë koste van vliegwiele is die relatiewe lae digtheid waarby energie geberg kan word, en hierdie studie ondersoek die redes hiervoor.

Die ontwerpmetodiek wat in die beskikbare literatuur voorgestel is, word krities geanaliseer en 'n paar verbeteringe word aanbeveel. Mees noemenswaardig is die opmerklike verbeteringe in energie-digtheid wat soms moontlik is indien die optimerings-probleem deurdag geformuleer word.

Omdat materiaaleienskappe 'n bepalende invloed op energie digtheid uitoefen word die probleem van materiaalseleksie ook verder bespreek. 'n Paar riglyne vir die seleksie van 'n optimale stel materiale sonder om oordrewe berekenings-inspanning te veroorsaak, word aan vliegwielontwerpers gegee. Hierdie voorstelle kan hopelik in die toekoms verder deurgetrap word as onderwerp vir verdere studies.

Sleutelwoorde: Hernubare energie, energie berging, vliegwiele, vliegwiel rotors, energie digtheid

Acknowledgements

The idea for this study originated several years ago while travelling in Africa. At one stage I found myself working for a company that sold power-generation equipment in a very under-developed part of the continent where, at the time, there was absolutely no electricity grid. The company mainly serviced the needs of the many non-governmental organizations and charities in the area, and amongst other things we often imported sets of solar panels for them. These presented quite an elegant solution for powering small offices or houses, because the only other alternative was diesel generators, which are noisy, expensive to operate and require regular maintenance.

However, there was a drawback to using solar panels, in that the sun couldn't always be counted on to shine exactly when you needed it. This necessitated the use of lead-acid batteries for storage, and even for small installations such as homes or offices it surprised me how bulky the battery banks needed to be in order to ensure dependable energy supply.

Later, while studying as an undergraduate, I realized that the same problem facing those small homes and offices also confronts the world's big utility companies when they try to incorporate renewable energy onto the electricity grid. In the absence of energy storage, a conventional electricity grid can really only accommodate very small amounts of renewable energy effectively.

After reading up on the available energy storage technologies and discussing the subject over some beers with other engineering students my thoughts eventually started converging on flywheels and the limits to the energy densities that they can achieve. It was early in 2008 that I first discussed the topic with Professor Detlev Kröger, who was kind enough to listen to my ideas. He encouraged me to pursue the topic further, and so I took it up as a final-year's project under the supervision of Derren Wood.

The outcome of the project was good, and so as postgraduate I had the opportunity to carry on in the same line of research, now under the auspices of Professor Albert Groenwold. I greatly appreciate the patient introduction into the world of academic research that he has given me, and I have to thank him, Derren Wood and Professor Kröger for their kind words of guidance that culminated in the timely completion of this study.

Then I would also like to thank Dr David Harley for proofreading parts of my work, Gene Hunt from Beacon Power Corporation for permission to use some of their images, and Laura from Renaissance Art Shop in Stellenbosch for supplying the recycled paper on which the original manuscripts were printed.

This study was funded by grants from the National Research Foundation as well as Denel through the HYSTOU research project.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This chapter provides a brief introduction into the workings of flywheels as energy storage devices. The importance of achieving high energy density is highlighted within the broader context of electrical energy storage. The scope for improving the energy density of flywheel rotors is then discussed.

Why energy storage? By many accounts renewable energy will have an important part to play in the electricity grid of the future. Fossil fuels are in limited supply, and evidence is mounting of its detrimental effect on our environment. This forces us to rethink the way we supply energy.

The generation of renewable energy has been shown to be possible, but now we are faced with another problem: Most renewable sources of energy are inherently intermittent. This raises the question of how to marry a fluctuating energy supply with a rather indifferent consumer demand.

Energy storage helps to alleviate this problem by providing a mechanism by which surplus energy can be stored during periods of low demand and supplied back to the grid when needed. When energy storage is implemented with renewable resources it decreases the intermittency of supply and adds value to the overall system [4].

Why flywheels? Several different storage concepts have been proposed, including the use of compressed air, superconducting magnets and chemical batteries. These new technologies are intended to supplement the existing pumped hydro storage facilities, for which future development is hampered by environmental concerns and a limited supply of suitable locations [5].

The use of flywheels poses several benefits over other storage technologies. These include high power density, long cycle lifetime and low environmental risk. How-

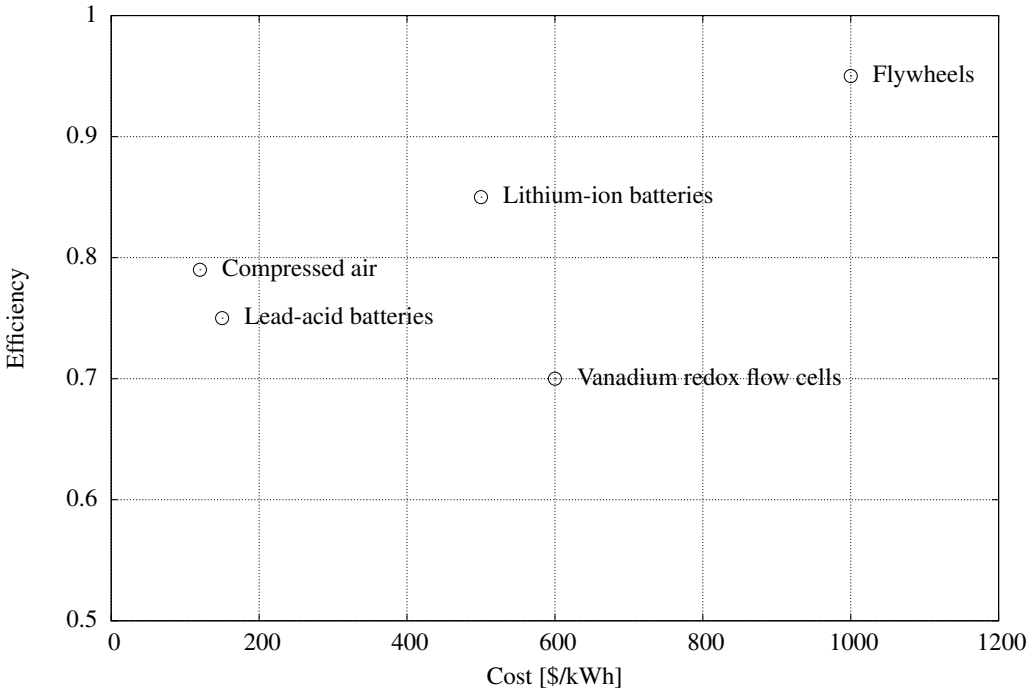


Figure 1.1: Efficiency and cost for various electrical energy storage technologies. Values are taken from a SANDIA report [6], where they are listed under *Distributed Generation Applications*.

ever, the most attractive attribute of flywheels at present is the high efficiency at which energy can be stored.

Most forms of electrical energy storage rely on some sort of energy conversion process, because electrical energy cannot be stored directly (save for capacitors and superconducting magnets). For flywheels this conversion is done by an electrical motor, which represents one of the most efficient ways of converting energy. For this reason flywheels can have overall efficiencies of around 95%, while most other technologies operate at efficiencies somewhere between 60% and 80%.

Figure 1.1 shows a comparison of the efficiency and cost of some storage technologies.

Why composites? Flywheels have been demonstrated to be viable for application in several niche markets. However, for the technology to be used on a wider scale, such as for the integration of renewables, it is necessary to address the high cost currently associated with flywheel systems. A big driver of cost in a flywheel system is the energy density in the rotor, which influences both material- and manufacturing costs and also the overall footprint of a storage facility.

The energy density of a flywheel rotor is largely determined by the strength and density of its constituting materials. For this reason, composite materials are ideal for use in flywheel rotors. Their high strengths and low densities allow for big improvements in energy density when compared to conventional steel rotors. In addition, the variable material properties of composite materials allow for tailorable designs.

How does it work? The concept of storing kinetic energy in a rotating mass is relatively simple, and has indeed been around for a very long time. Archaeological evidence suggest that flywheels were already commonly used in potters' wheels around 3000 years BC [7].

Whereas the potters of old had to spin their flywheels up by hand, the type of machines that we are interested in takes electrical energy as input. A typical flywheel is shown in Figure 1.2. It consists of a cylindrical rotor, which is connected to a shaft by a hub. The shaft is driven by an electrical motor which converts electrical energy to kinetic energy and back. Magnetic bearings are used to keep friction losses to a minimum, and the whole assembly is housed in an enclosure. This may serve a dual purpose by maintaining a vacuum over the moving parts while also containing fragments in the case of a rotor failure.

The kinetic energy of a spinning rotor is given by

$$E = \frac{1}{2} I \omega^2, \tag{1.1}$$

where E is the kinetic energy, ω is the rotational velocity and I is the rotor's moment of inertia, given by

$$I = \int_V \rho r^2 dV. \tag{1.2}$$

Here, r is the distance from any point to the axis of rotation, ρ is the density of the rotor's material and the integral is evaluated over the volume V of the rotor.

1.1 Theoretical limits

For a given material, the upper bound to the energy density that can be achieved in a flywheel rotor is given by the ratio of its tensile strength X to density ρ. For this reason, fiber-composite materials are by far superior to their isotropic counterparts when it comes to storing kinetic energy. This can be seen in Table 1.1 where some indicative values are given for the upper limit to the energy density of a few interesting materials. However, for a flywheel to store energy at such a high density, it would have to be perfectly uniformly stressed under centrifugal loading. To the

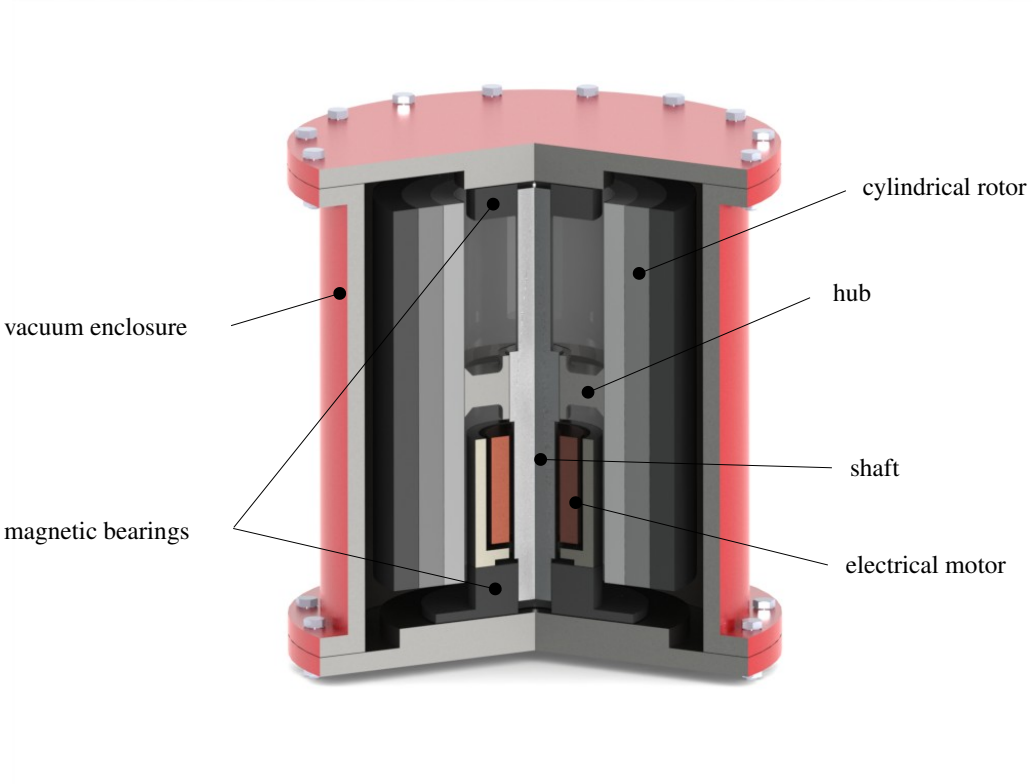


Figure 1.2: An example of a typical flywheel assembly.

Material	$\left(\frac{X}{\rho}\right)$ [Wh/kg]
Isotropic	
Aluminium	61.7
Steel	70.6
Titanium	80.0
Fiber-composite	
S-glass/Epoxy	250
HM Graphite/Epoxy	187
HS Graphite/Epoxy	371
Kevlar/Epoxy	403

Table 1.1: Strength-to-density ratios of some interesting materials. These represent upper bounds to the energy density that can be achieved in rotors made from these materials. For fiber-composites a unidirectional layup is assumed and X represents the tensile strength along the fiber direction.

best of our knowledge, this can only be achieved in the hypothetical *Constant Stress Disc* profile [8], and only for isotropic materials.

For the fiber composite materials that we are interested in, the best possible geometry seems to be that of a thin cylinder. Under centrifugal loading, such a rotor is uniformly stressed in the circumferential direction, and completely unstressed in the radial direction. For this geometry, the energy density of a rotor is exactly one half of the strength-to-density ratio of its constituting material.

This is still a very high value, and practical cylindrical rotors deviate from this ideal because they are rarely very thin in the radial direction. Thick rotors are desirable for practical applications because of the need to store some finite amount of energy. For this purpose the rotor not only needs a high energy density, but it also needs to be sufficiently massive.

Two factors tend to limit the energy density of practical cylindrical rotors: firstly, they experience non-uniform stress distributions in the circumferential direction. This causes some stress concentration to limit the design and keep the rest of the rotor from becoming fully stressed. Secondly, the rotors do experience stress in the radial direction, and even though the values of these stresses may be relatively low, they act in the material's weakest direction. Such stresses can cause a rotor to fail by delamination while it is still far from being fully stressed in the circumferential direction.

1.2 Examples from industry

A good example of the type of flywheel that we are concerned with in this study is the *Smart Energy 25* flywheel recently developed by Beacon Power Corporation in the USA. Figure 1.3 shows the composite rotor, made from carbon- and glass fibers being lowered into its vacuum housing. This rotor is designed to store 25 kWh of retrievable energy.

The assembled flywheel, as shown in Figure 1.4, is designed to deliver large bursts of energy at very short notice. These flywheels are rated at 100 kW, and they can deliver this continuously for a maximum of 15 minutes.

In order to minimize the effects of rotor failure, these flywheels are installed in concrete enclosures, under ground level. Figure 1.5 shows a flywheel being lowered into place at Beacon's Stephentown facility in New York. The purpose of this facility is to provide frequency regulation to the New York electricity grid, which makes it the first grid-scale flywheel energy storage facility in the world. Construction at the site started in earnest in May 2010.

In Figure 1.6 a group of flywheels can be seen, installed around a container which houses the power-electronics necessary to operate them. The Stephentown site is divided into clusters consisting of ten flywheels each, and rated at 1 MW.

The overall facility is shown in Figure 1.7. It consists of 200 flywheels and was completed only recently, with the inauguration ceremony being held on the 12th of July of this year. It boasts a power rating of 20 MW, which can be supplied continuously for a maximum of 15 minutes. This gives an overall energy storage rating of 5 MWh.

1.3 Thesis outline

In this thesis a critical analysis is made of the methods that are available for the design of high energy density flywheel rotors. Towards this aim an overview of the available literature is presented in the following chapter and the design problem is introduced.

This is followed in Chapter 3 by a discussion on the use of a geometric shape factor as a measure of the energy density of a rotor. In Chapter 4 this shape factor is used when making an in-depth analysis of the different ways that the design optimization problem is formulated in literature. A novel formulation is proposed, which overcomes some of the difficulties encountered in the work of other researchers. The use of this new formulation is shown to lead to considerable improvements in energy density in certain cases.

Figure 1.3: A cylindrical composite rotor being lowered into its vacuum housing. Image courtesy Beacon Power Corporation.

Figure 1.4: A fully assembled flywheel. Image courtesy Beacon Power Corporation.

Figure 1.5: A flywheel assembly being lowered into its concrete enclosure. Image courtesy Beacon Power Corporation.

Figure 1.6: Clusters of flywheels installed around the containers that house their power electronics. Image courtesy Beacon Power Corporation.

Figure 1.7: Aerial view of the completed 20 MW, 5MWh energy storage facility at Stephentown, New York, USA. Image courtesy Beacon Power Corporation.

In Chapter 5, the problem of material selection is discussed. Some new ideas are put forward, and opportunities for further research are highlighted. Thereafter, in Chapter 6, the methods of analysis that were used during this study are outlined.

Ultimately, Chapter 7 provides a conclusion and a summary of recommendations for the design of flywheel rotors for high energy density.

Chapter 2

Literature review and problem statement

In this chapter, a summary is provided of the most relevant literature that is available in the field of flywheel rotor design. Also, an outline is given to the problem of designing rotors for high energy density.

2.1 Context

Much of the groundwork for the study of flywheel rotor optimization was laid during the 1980's by Giancarlo Genta. His text [9] on flywheel energy storage arguably remains one of the best cited publications in this field.

In his work, Genta considers a wide variety of flywheel rotor geometries. Some interesting examples include bare filament rotors [10], cylindrical rings [11] and profiled discs. For the latter, both steel discs [8] and composite discs [12] are evaluated.

In much of Genta's work, the *modus operandi* is to first find the best possible material for a given application. Usually such a material will be required to have a very high ratio of strength to density. After the material is selected, the design problem is to find the most suitable geometry; one that would result in a stress state as close to uniform as possible at the point of failure. Around this theme, further research has been undertaken to study the additional improvements that could be made by varying material properties through a rotor or inducing a certain residual stress state in some way or another.

Other authors have also proposed some very interesting rotor concepts. The cylindrical rotor proposed by Danfelt [13] is one example. It includes thin layers of

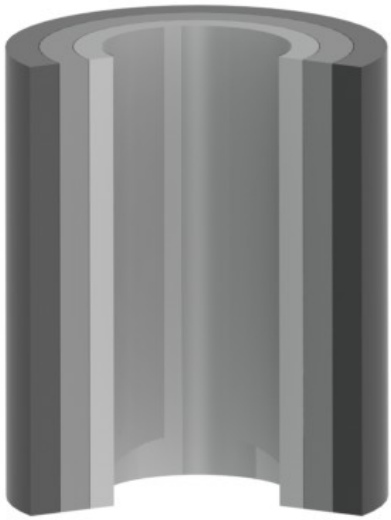


Figure 2.1: An example of a layered cylinder.

rubber in-between concentric layers of fiber composites in order to manipulate the radial stress distribution in the rotor. The topic has recently been carried forward by Portnov [14], although this approach does not seem to have found practical application in industry. In another recent work [15], Fabien studies cylindrical rotors with some of the fibers oriented in the radial direction. These rotors have also not been widely applied in industry, probably due to the difficulty of manufacturing them.

The type of rotors that are being adopted with greatest success are those that consist of simple concentric cylinders of a few different composite materials. Such rotors can be found providing bridging power for large backup UPS installations, controlling voltage on electrical rail networks, and lately also regulating frequency on the electricity grid. An example of such a rotor is shown in Figure 2.1.

Notable research in this field includes that done by Sung Kyu Ha, who developed an analytical model for analysing the stress distribution inside a cylindrically wound composite rotor [3]. He showed how this model could be used when searching for an optimal distribution of material layers inside a rotor and also highlighted the importance of considering residual stresses from curing in such an analysis.

An important contribution, for which Andrew Arvin [2] gives the credit to Genta [9] and Portnov [16], was made by noting that it can be advantageous to arrange materials in a cylindrical rotor in such a way that the specific stiffness increases with

radius. In fact, this is routinely found in practice and, in line with the methods of other authors such as Krack *et al.* [1], this study *disregards material sequences that do not result in increasing specific stiffness with radius*. However, there does not seem not be any rigorous proof that this approach is always advantageous, and further research may therefore be warranted in future.

Ha made another important contribution by showing that an optimal design, when scaled geometrically, remains optimal [17]. This is of great use, as it implies that a design problem can be solved to find a whole family of optimal rotors, from which a designer can choose one that suits the scale of the particular application.

Most recently, Pérez-Aparicio has proposed another analytical model for evaluating stress in composite cylinders [18]. This model aims to take into account the effect of a hub while also providing for the effect of non-uniform curing and moisture absorption during the lifetime of the composite part.

2.2 Design problem

In order to increase the energy density of practical cylindrical flywheel rotors, it is necessary to find designs that are closer to uniformly stressed under centrifugal loading.

This may be done in two ways: Firstly, the stress distribution is influenced by material density and stiffness, so by making use of more than one material within a given rotor, the material distribution may be used to manipulate the stress distribution to some extent.

Secondly, it is possible to include a certain level of pre-stress within a rotor, usually by press-fitting layers of material on to each other. If this is done intelligently it may serve to reduce local stress concentrations and hence increase the energy density of a rotor.

The second approach certainly works, and it is widely used in industry, but it does add to the complexity of the manufacturing process which is thought to be one of the big contributors to the high cost of flywheel systems. If at all possible it would be desirable to have high energy density designs that do not need to use this approach.

The first approach is illustrated in Figure 2.2, which depicts the stress distribution in a thick single-material rotor at the point of failure. It can be seen that the radial stress σ_r limits the design even though the circumferential stress σ_c is very low. Such a rotor is far from fully stressed and does not achieve a very high energy density. On the other hand, Figure 2.3 shows the stress distribution at the point of failure in a rotor of the same thickness, but consisting of three distinct material layers. It can be seen that the rotor fails simultaneously in several places and that it is generally

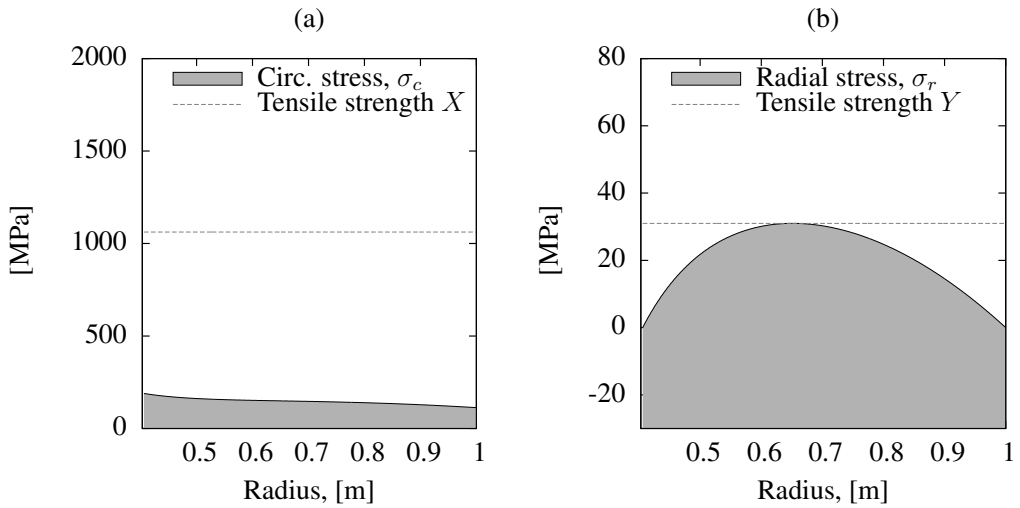


Figure 2.2: Stress distribution in a thick single-material rotor at the point of failure: (a) Circumferential stress along the fiber direction. (b) Radial stress across the fiber direction.

closer to fully stressed. Such a rotor can achieve a much higher energy density than its single-material counterpart.

In conclusion, the problem facing a flywheel rotor designer is to first find the most suitable materials, and then to find the best possible arrangement for those materials within a rotor, keeping in mind the effect that material distribution has on the stress experienced under centrifugal loading. Figure 2.4 shows a section of a flywheel rotor with three material layers. For such a rotor the designer needs to determine suitable values for the thickness of each layer, as well as the overall thickness of the rotor in order to ensure optimal energy density when the rotor is operating at its maximum rotational velocity.

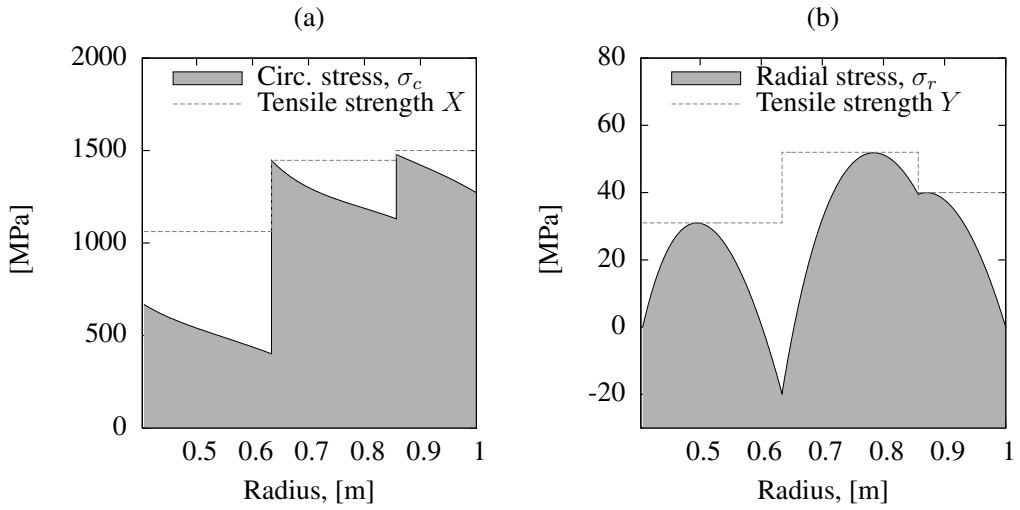


Figure 2.3: Stress distribution in a thick rotor made from three materials at the point of failure: (a) Circumferential stress along the fiber direction. (b) Radial stress across the fiber direction.

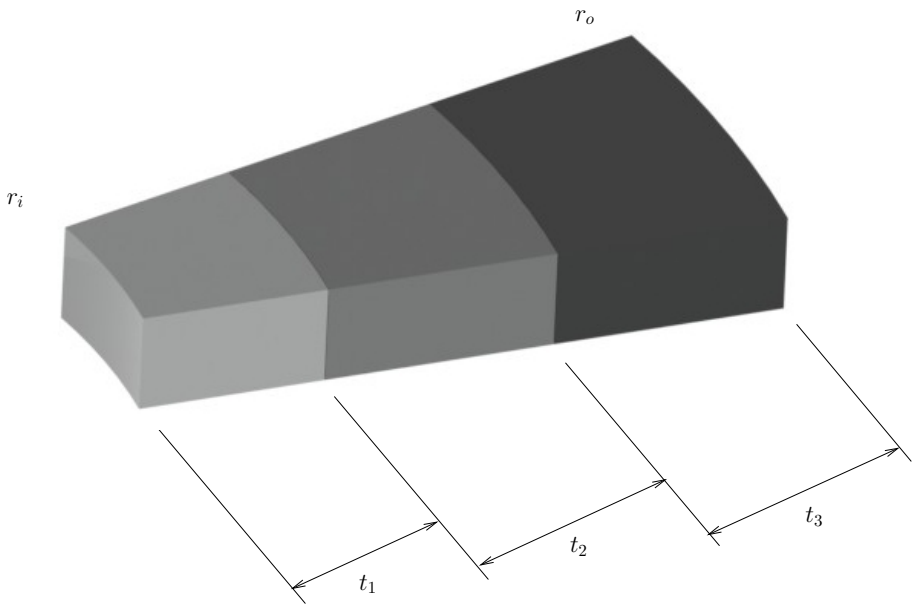


Figure 2.4: Section of a layered cylindrical rotor.

Chapter 3

Shape factor as measure of energy density

The energy density of a particular rotor not only depends on the suitability of the chosen design parameters, but also on the merits of the material properties that were made available for the design. In this chapter we introduce a dimensionless geometric shape factor. This parameter can be used in order to strike a comparison between rotor designs that were based on different material sets.

When considering the suitability of a particular rotor for a given application, the energy density of the rotor is usually of great importance, and decisions may be made based on the value of this parameter. However, when comparing different methods of rotor design, care should be taken not to compare the energy density of rotors that were not based on the same set of materials. This is because material properties greatly influence the energy density that a rotor may be able to achieve, and so a relatively poor design method may still yield a better energy density if better materials are available.

In order to strike an objective comparison between different rotor designs, it is necessary to determine the energy density of a rotor relative to the potential of the materials that were used. For this purpose, Genta makes use of a dimensionless parameter known as the shape factor K, which he defines as [9]

$$\frac{E}{m} = K\left(\frac{\sigma}{\rho}\right), \tag{3.1}$$

where E is the kinetic energy of the rotor, m is the mass, σ is the material strength and ρ is the material density.

In its simplest form the shape factor is the weight energy density of the rotor divided by the material's strength-to-density ratio. This gives an indication of how well a

Geometry	Shape Factor, K
Constant stress disc, theoretical	1
Constant stress disc, practical	0.645
Constant thickness disc, unpierced	0.606
Thin ring, without hub	0.5
Rod, constant thickness	0.333

Table 3.1: Value of the shape factor for some common geometries.

rotor exploits the energy storage potential of the materials from which it is made. The shape factors of some common geometries are shown in Table 3.1. In deriving these values the implicit assumption is made that the constituent materials are isotropic in nature, so that the material strength is the same in all directions.

Unfortunately, the shape factor as defined in Equation (3.1) cannot be applied directly to the cylindrical composite rotors that we are interested in. This is because they are not isotropic, and they are usually not made from a single material. Perhaps for these reasons, the shape factor has fallen from use recently. Notably, it is not given any mention in the three articles [2, 3, 1] that will be used in the next chapter when evaluating the design methods used in literature.

In this chapter the concept of a geometrical shape factor is extended to account for rotors made from multiple orthotropic materials.

3.1 Shape factor for non-isotropic materials

In deriving the shape factors for different geometries in Table 3.1 according to the definition in Equation (3.1), the implicit assumption is made that the maximum stress inside the rotor limits the design, or that the value for strength used in Equation (3.1) is applicable in the direction where failure will first occur.

Indeed, for isotropic rotors this is the case, because the strength of the material is the same in all the principal directions. However, for composite rotors the strength in these directions may differ greatly.

The energy density of a cylindrically wound, composite rotor will be greatest if the rotor is fully stressed in the direction of greatest strength. For this reason it makes sense to relate energy density to the strength-to-density ratio by defining the shape factor as

$$\frac{E}{m} = K\left(\frac{X}{\rho}\right), \tag{3.2}$$

where E is the kinetic energy of the rotor, m is the mass, X is the material strength in tension along the fiber direction and ρ is the material density.

It should be noted that the values of K given in Table 3.1 may or may not apply to composite rotors, depending on the choice of fiber direction within the rotor.

3.2 Shape factor for multiple materials

The definition for the shape factor, as given above, also does not apply when more than one material is present in a rotor, because there are several different possible values of strength and density to consider. However, we propose that a weighted average shape factor may have the same meaning for a rotor made from several materials as the usual shape factor does for a single material rotor. The new definition is stated as

$$K = \sum_{i}^{n} \left(\frac{m_i}{m}\right) K_i, \tag{3.3}$$

where the subscript i refers to the i^{th} layer of material and n different materials are considered. The shape factor for an individual layer is defined in Equation (3.2) to be

$$K_i = \frac{E_i}{m_i} \left(\frac{X}{\rho}\right)_i^{-1}. \tag{3.4}$$

Here, E_i is the kinetic energy of layer i, m_i is the mass of layer i and $\left(\frac{X}{\rho}\right)_i$ is the strength-to-density ratio of the constituting material of layer i, expressed in terms of the tensile strength along the fiber direction, X.

3.3 Theoretical limit

For the composite cylinders that we are interested in, the highest possible value for the shape factor is achieved when the rotor is thin in the radial direction. In this limiting case the inside- and outside radii are approximately equal and the value for the shape factor $K = 0.5$. At the point of failure, such a rotor is fully stressed in the circumferential direction.

A design that is able to achieve a shape factor of 0.5 represents an upper bound to the energy density that can be achieved with a given set of materials. If a rotor has a shape factor that is far from 0.5, then this gives an indication that the energy storage potential of the material set is not fully exploited.

In the next chapter some design problems from literature are analyzed and the shape factor is used to show how well these designs exploit the material properties at their disposal.

Chapter 4

Rotor design for optimal energy density

In searching for optimal designs, the way in which the optimization problem is formulated was found to be of particular importance. In this chapter, some formulations that were encountered in literature are analyzed in detail and a novel formulation is then proposed which is shown to lead to significant improvements in energy density in certain cases.

At present there is no consensus on how an optimal flywheel rotor should be designed [2]. Authors choose to formulate the optimization problem in different ways and the approaches found in literature differ greatly in choice of objective function, constraints and design variables. Furthermore, the effects of these differences are unclear, because problems are usually based on different material sets and results cannot be compared directly.

Some popular choices of objective function include weight energy density [2, 19], cost [1] and total stored energy [3]. In some of these cases the designs are constrained by prescribing a fixed rotational velocity [3], in others the design is limited by prescribing a fixed ratio of outside- to inside radius [1]. The reasons for these differing approaches possibly has to do with the fact that authors want to address different applications, but it is still possible to assess how well the resulting flywheel rotors make use of the energy storage potential of the materials that they are built from.

In this chapter the effect of using these different problem formulations is investigated by calculating the energy density and shape factor for some of the optimal rotors that have previously been reported. A new problem formulation is then proposed and it is shown how the use of this problem formulation can sometimes lead to dramatically improved results.

Property	Description	Glass/epoxy	Carbon/epoxy	Units
E_1	Young's modulus	41	147	[GPa]
E_2		10.4	10.3	[GPa]
G_{12}	shear modulus	4.3	7	[GPa]
ν_{12}	Poisson's ratio	0.28	0.27	
X	tensile strength	1140	2280	[MPa]
X'	compressive strength	620	1725	[MPa]
Y	tensile strength	39	57	[MPa]
Y'	compressive strength	128	228	[MPa]
ρ	density	1970	1600	[kg/m^3]
X/ρ	strength-to-density ratio	160.7	395.8	[Wh/kg]

Table 4.1: The set of materials used by Krack *et al.* [1]. For the material strength parameters, X corresponds to the fiber direction, and Y to the direction normal to the fiber direction.

The content of this chapter is largely based on an article submitted for publication, which in turn was based on work presented at the African Conference on Computational Mechanics in Cape Town [20].

4.1 Design approaches from literature

In this section some recently used formulations for designing flywheel rotors are discussed. The three example problems that are used are all based on different material sets. In order to strike a comparison between the resulting optimal designs it is therefore necessary to calculate the shape factor of these rotors.

The design approaches that are discussed here differ in choice of objective function, constraints and design variables, so some telling differences can be seen in the results. The lessons learned from this section are incorporated into a suggested problem formulation which is introduced in the next section.

For each example problem the details of the original optimal rotor is given, together with the set of material properties used by the authors. The nature of the chosen problem formulations are discussed and the energy density and shape factor is calculated for each rotor.

4.1.1 Example 1: *Krack et al.*

In a recent example, Krack *et al.* [1] consider the design of a two-material rotor with the material cost of the rotor taken as the objective function. This objective is similar to weight energy density, but an additional weighting-parameter is assigned

	Results	Units
Layer thickness	$t_1 = 47.61$, (Glass/epoxy)	[mm]
Rotational velocity	46711	[rpm]
Axial thickness	120	[mm]
Inner radius	120	[mm]
Outer radius	240	[mm]
Total stored energy	3252.3	[Wh]
Volumetric energy density	149775	[Wh/m^3]
Weight energy density	116.29	[Wh/kg]
Shape factor, K	0.39	

Table 4.2: The resulting optimal design reported by Krack *et al.* [1].

to each material by defining a cost ratio. The set of material properties that are used are repeated in Table 4.1.

Several optimal rotors are reported for different failure criteria and material cost ratios. As a reference case we choose the result where the Tsai-Wu failure criterion is applied and the cost ratio is unity. For this cost ratio the objective function is equivalent to weight energy density.

The optimization problem is formulated in such a way that only two design variables are considered, namely the thickness of one of the material layers and the rotational velocity. Fixed values are prescribed for the inner- and outer radius of the rotor and also for the rotor's axial thickness. The rotor is constrained to have a maximum strength ratio of less than unity.

Stresses are evaluated by means of an analytical plane stress model, which is based on the work of Ha *et al.* [21]. For this model the effect of stress in the axial direction is not considered, and it should therefore only be used for axially thin rotors. Residual stress from curing is not considered by the authors, and interference fits are not allowed.

The rotor's strength ratio is calculated according to the quadratic Tsai-Wu failure criterion, described in Chapter 6.

Details of the resulting rotor with optimal weight energy density are given in Table 4.2. When the rotor's shape factor is calculated, the value is found to be $K = 0.39$. This is a reasonably good result, but further improvement is possible. After closer inspection it becomes evident that the performance of the formulation is highly problem-specific. The objective function is strongly influenced by the choice of inner- and outer radius, but the authors have apparently selected these values arbitrarily. In Section 4.3 it is shown that improvements in energy density can be made by formulating the problem in a different way, which takes all possible values for the inner- and outer radii into account.

Property	Description	T700	T1000	M46	Units
E_1	Young's modulus	148	195	278	[GPa]
E_2		7.8	7.5	6.9	[GPa]
ν_{12}	Poisson's ratio	0.34	0.30	0.30	
X	tensile strength	1450	1800	1280	[MPa]
X'	compressive strength	928	928	579	[MPa]
Y	tensile strength	50	50	50	[MPa]
Y'	compressive strength	70	70	70	[MPa]
α_1	thermal expansion coeff.	0.7	0.7	-0.4	$[10^{-6}/°C]$
α_2		35	35	35	$[10^{-6}/°C]$
ρ	density	1570	1570	1590	[kg/m^3]
X/ρ	strength-to-density ratio	256.5	318.5	223.6	[Wh/kg]

Table 4.3: The set of materials used by Arvin and Bakis [2]. For the material strength parameters, X corresponds to the fiber direction, and Y to the direction normal to the fiber direction.

	Results	Units
Inner- and outer radii of each layer	$r_{i1} = 9.273$, $r_{o1} = 11.890$, (T700) $r_{i2} = 11.886$, $r_{o2} = 15.751$, (T1000) $r_{i3} = 15.700$, $r_{o3} = 21.065$, (T1000) $r_{i4} = 21.005$, $r_{o4} = 24.786$, (M46) $r_{i5} = 24.754$, $r_{o5} = 26.817$, (M46)	[mm]
Temperature difference	-112	[°C]
Rotational velocity	298893	[rpm]
Volumetric energy density	76325	[Wh/m^3]
Weight energy density	54.9	[Wh/kg]
Shape factor, K	0.22	

Table 4.4: The resulting optimal design reported by Arvin and Bakis [2], with the inner layers of steel and magnets not included in the calculation of the energy density or shape factor.

4.1.2 Example 2: *Arvin and Bakis*

In another example, Arvin and Bakis [2] investigate the design of a flywheel rotor intended for application in a small satellite. A set of three materials are used, and their properties are reported in Table 4.3. For this problem the composite rings are prescribed to fit on to an inner-ring of steel and magnets, which forms part of the motor/generator of the flywheel. Herein we are only interested in the performance of the composite rings.

The weight energy density of the rotor is used as objective function, while rotational velocity, the number of rings and the inner- and outer radius of each ring are taken as design variables. The design is constrained by prescribing a fixed value for the inner radius, and by enforcing a maximum strength ratio of less than unity.

Property	Description	Glass/epoxy	T300/2500	T800H/2500	Units
E_1	Young's modulus	38.6	130	155	[GPa]
E_2		8.27	9	9	[GPa]
G_{12}	shear modulus	4.14	4.55	4.55	[GPa]
$\nu_{12} = \nu_{21}$	Poisson's ratio	0.26	0.30	0.30	
X	tensile strength	1062	1800	2900	[MPa]
X'	compressive strength	610	1400	1600	[MPa]
Y	tensile strength	31	80	70	[MPa]
Y'	compressive strength	118	168	168	[MPa]
S	shear strength	72	48	48	[MPa]
α_1	thermal expansion coeff.	8.6	-0.3	-0.3	$[10^{-6}/°\text{C}]$
α_2		22.1	28.1	28.1	$[10^{-6}/°\text{C}]$
ρ	density	1800	1600	1600	[kg/m^3]
X/ρ	strength-to-density ratio	163.9	312.5	503.5	[Wh/kg]

Table 4.5: The set of materials used by Ha *et al.* [3]. For the material strength parameters, X corresponds to the fiber direction, and Y to the direction normal to the fiber direction.

Again an analytical plane stress model is used for the stress analysis, which is accurate for very thin rotors. This model allows for interference fits between rings. The residual stress from curing is also considered.

In order to analyze rotor strength, the author requires that three failure criteria be simultaneously satisfied. The first is simply the maximum stress criterion, which is described in Chapter 6. The second and third failure criteria are variations of the quadratic Tsai-Wu criterion, for which the F_{12}^*-parameter from Equation (6.32) is set to -0.5 and 0 respectively.

The resulting optimal design is repeated in Table 4.4. The shape factor for this rotor is calculated to be $K = 0.22$, which does not compare well to the theoretical maximum of 0.5, and so the design does not come close to utilizing the full energy storage potential of its materials.

The low energy density of the reported result is mostly due to the choice of failure criteria. In particular, the variant of the Tsai-Wu criterion which uses a constant value of $F_{12}^* = 0$ is found to be very restrictive, and it is not clear what the motivation is for using this. However, even with all three failure criteria in place a significant improvement in energy density is possible with the set of materials used in this problem. This is demonstrated in Section 4.3.

4.1.3 Example 3: *Ha et al.*

In the final example, Ha *et al.* [3] were the first to note the importance of considering residual stress from curing when designing optimal flywheel rotors. Their

	Results	Units
Layer thicknesses	$t_1 = 28.26$, (Glass/epoxy) $t_2 = 38.14$, (T300/2500) $t_3 = 38.82$, (T800H/2500)	[mm]
Temperature difference	-100	[°C]
Axial thickness	100	[mm]
Outer radius	155.22	[mm]
Rotational velocity	60 000	[rpm]
Total stored energy	794	[Wh]
Volumetric energy density	104900	[Wh/m^3]
Weight energy density	71.7	[Wh/kg]
Shape factor, K	0.18	

Table 4.6: The resulting optimal design reported by Ha *et al.* [3].

paper considers a set of three materials for which the properties are given in Table 4.5.

In formulating the optimization problem, total stored energy is chosen as the objective function. The value for the axial thickness and the inner radius of the rotor are fixed, as is the value of the rotational velocity. The thicknesses of each of the three layers are the only design variables. The design is constrained to have a maximum strength ratio of less than unity.

For stress evaluation an analytical model is proposed which is based on the plane strain assumption, but which allows for strain in the axial direction to vary through the thickness of the rotor. No interference fits are allowed and, in order to allow for residual stress caused by curing, an initial temperature difference is prescribed.

The quadratic Tsai-Wu failure criterion is used, as given in Chapter 6.

The resulting optimal design is presented in Table 4.6. This rotor's shape factor is calculated to be $K = 0.18$, which is much lower than the theoretical limit of 0.5. This shows that the resulting rotor does not utilize the energy storage potential of its constituting materials very well at all.

The low energy density and shape factor of the optimal rotor can be attributed to the constraint placed on rotational velocity. This causes some candidate rotors to be evaluated far from their point of failure. In other words, the total stored energy that is used for comparison is not the maximum possible total stored energy of the candidate rotor, but rather the total stored energy at the specified rotational velocity. When this constraint is lifted, considerable improvement in energy density, and total stored energy is possible. Again, this is demonstrated in Section 4.3.

4.2 Proposed formulation

As seen in the previous section, the problem formulations used in the literature do not consistently lead to rotors with high energy density. Based on the high strength-to-density ratios of the materials that are used, the energy density of some of these rotors are surprisingly low.

In this section, a new problem formulation is proposed for which the resulting optimal rotors will consistently achieve high energy density relative to their constituting materials.

4.2.1 Design variables

Before designing a cylindrical composite rotor with several layers of different materials, it is instructive to consider the design of a single material rotor. For such a rotor the design problem is only to find the most suitable values for the inner- and outer radii. In this case it is tempting to take the absolute thickness of the rotor as design variable. However, for energy storage purposes, rotors that are geometrically similar have identical energy storage characteristics [17]. For this reason a measure for the relative thickness of a rotor can be taken as the design variable. One possibility is the ratio of inside- to outside radius $\left(\frac{r_i}{r_o}\right)$. When this design variable is used, the problem is solved in a more general fashion, because the result is not a single rotor, but rather a family of optimal rotors from which a rotor can be chosen which suits the scale of the problem at hand.

For rotors with multiple layers of materials the overall geometry can also be described by a relative thickness parameter. Furthermore, if the thickness of each individual layer of material is taken relative to the overall thickness, then the results of such a design problem are also scalable and a family of geometrically similar optimal rotors can be found. This is illustrated at the end of this section.

In summary, we suggest the use of the design variables

$$\left(\frac{r_i}{r_o}\right), t_1, t_2, ..., t_{n-1},$$

where t_i is the relative thickness of the i^{th} layer of material and the problem considers n different materials.

4.2.2 Constraints

The aim should be to add as few constraints as possible, so as not to artificially restrict the search domain. However, one important constraint is to limit the rotational velocity to its maximum value for any given rotor. In other words, all of the

candidate rotors should be compared at the point of failure, where they achieve the highest possible energy density. This allows all candidate rotors to be compared on an equal footing. Also, it ensures that the Tsai-Wu failure criterion can be used safely during the optimization. This is not necessarily the case when structures are analyzed far from the point of failure [22].

If necessary, the axial thickness of a rotor can be included as a constraint by specifying a value for the ratio of axial thickness h, to outer radius r_o. By specifying the axial thickness in such a relative way the results remain geometrically scalable. Additionally, boundary constraints are necessary to limit the design variables to sensible values.

With everything considered, we propose a constraint on the rotational velocity

$$\omega = \omega_{max},$$

where only the rotational velocity at failure is considered. This is the same as placing a constraint on the maximum strength ratio $R_{max} = 1$, which forces evaluation to take place at the point of failure.

Furthermore, boundary constraints should be placed on the values of the design variables:

$$0 < \left(\frac{r_i}{r_o}\right) < 1,$$
$$0 \leq t_i \leq 1,$$

and the sum of the layer thicknesses should not exceed the overall thickness of the rotor

$$\sum_{i}^{n-1} t_i \leq 1.$$

4.2.3 Objective function

When optimizing for maximum energy density, the choice of objective function can be either weight energy density, *i.e.* energy per unit mass, or it can be volumetric energy density, *i.e.* energy per unit volume. These objectives are plotted against the overall relative thickness for a representative single material rotor in Figure 4.1, where it can be seen that the weight energy density increases monotonically and the highest possible value is achieved for a thin rotor with $\left(\frac{r_i}{r_o}\right) = 1$.

Such a thin rotor is not of practical interest, because a rotor needs to be massive if it is to store any finite amount of energy. The peril in using weight energy density,

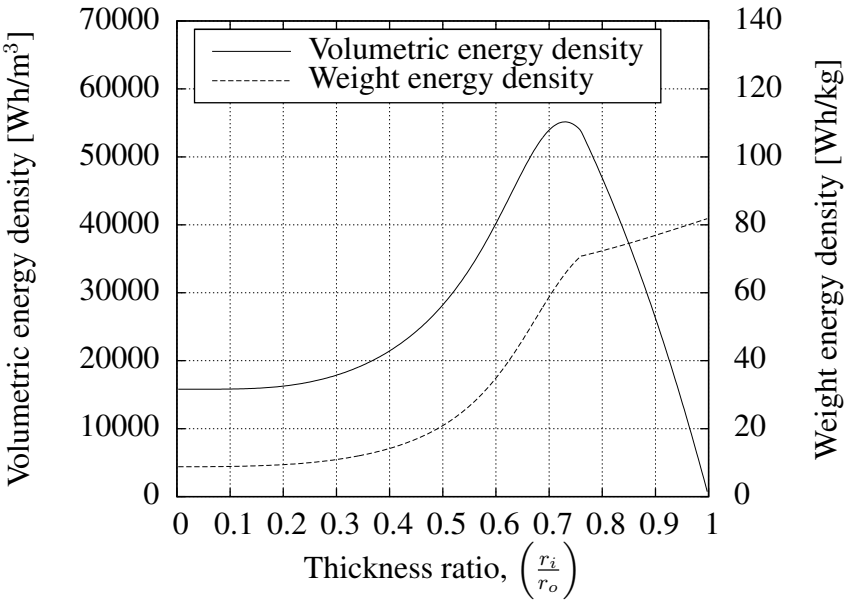


Figure 4.1: The objective function for a single material rotor design problem.

material cost or even the shape factor as objective function is that the result will be the thinnest possible rotor which satisfies the constraints, and this is not always desirable.

We therefore propose the use of volumetric energy density as objective function; *i.e.*

$$f = E_{vol},$$

where E_{vol} is the volumetric energy density, defined as the kinetic energy divided by the cylindrical volume enclosed by the outer radius.

For a single material rotor, the volumetric energy density is also depicted in Figure 4.1. By using the volumetric energy density as objective the result will usually be the thickest (*i.e.* most massive) rotor which still has a reasonably high weight energy density. Such rotors are of great practical interest.

4.2.4 Summary

For a representative two material problem the design space is shown in Figure 4.2. The only two design variables are the relative overall thickness of the rotor, $\left(\frac{r_i}{r_o}\right)$, and the relative thickness of the first layer of material, t_1. The plane stress assumption is made and the quadratic Tsai-Wu failure criterion is used.

Due to the nature of the problem formulation the results are scalable, and so the solution is not a single rotor, but rather a family of geometrically similar rotors. To illustrate this, a Pareto curve for the solution is plotted in Figure 4.3. Here

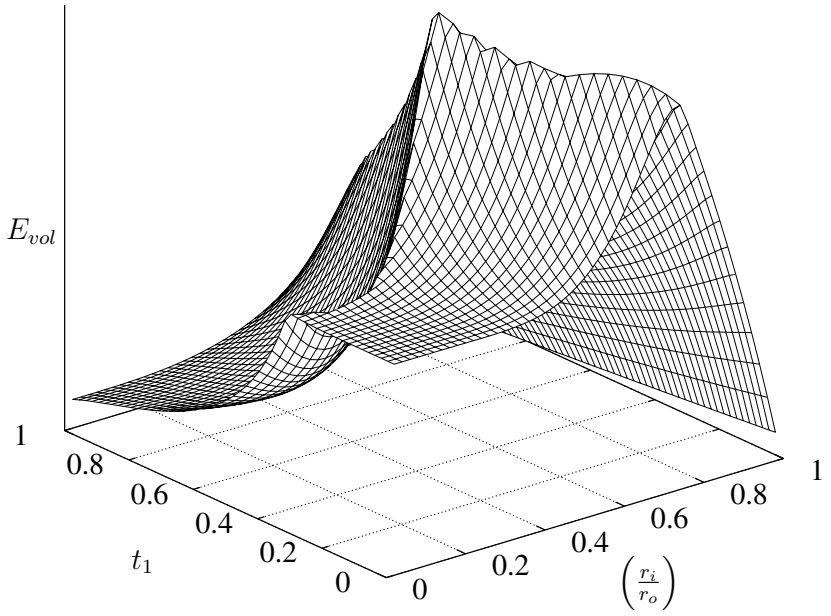


Figure 4.2: The design space for a representative plane stress problem considering two materials.

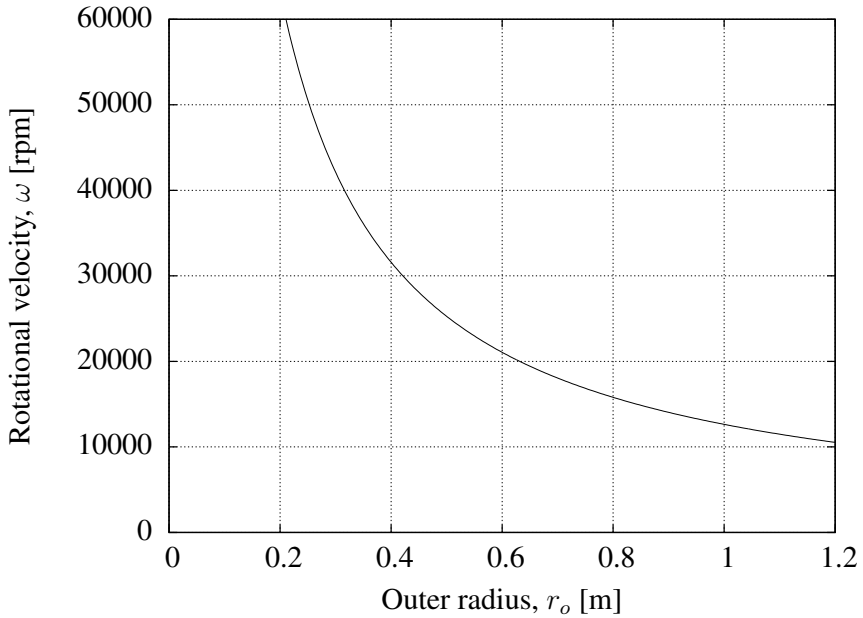


Figure 4.3: A Pareto front that represents the solution to the plane stress problem in Figure 4.2. Any point on this curve represents an allowable combination of rotational velocity and outer radius for an optimal rotor based on the given material set.

	Example 1	Example 2	Example 3	Units
Layer thickness	$t_1 = 62.84$, (Glass/epoxy)	$t_1 = 3.937$, (T700) $t_2 = 4.737$, (T1000)	$t_1 = 30.11$, (Glass/epoxy) $t_2 = 0.0$, (T300/2500)	[mm]
Overall geometry	$\left(\frac{r_i}{r_o}\right) = 0.462$	$\left(\frac{r_i}{r_o}\right) = 0.627$	$\left(\frac{r_i}{r_o}\right) = 0.471$	
Temperature difference	0.0	-112	-100	[°C]
Axial thickness	120	n.a.	100	[mm]
Outer radius	240	26.817	155.22	[mm]
Rotational velocity	47750	346448	80458	[rpm]
Total stored energy	3500.5	n.a.	1393.9	[Wh]
Volumetric energy density	161207	87543	184160	[Wh/m^3]
Weight energy density	117.4	91.6	142.9	[Wh/kg]
Shape factor, K	0.418	0.331	0.384	

Table 4.7: Resulting optimal designs when using material sets from [1], [2] and [3] together with the problem formulation proposed in this chapter.

every point on the curve represents an optimal rotor for a given maximum rotational velocity or outer radius.

4.3 Optimal rotors

In this section the use of the problem formulation described in Section 4.2 is demonstrated. Three different material sets are used for which optimal rotor designs have previously been published. In each case the new problem formulation is used to find a rotor with optimal volumetric energy density while using the same failure criteria suggested by the previous authors. The axisymmetric finite element method is used for stress evaluation. Each result is scaled to fit into the same cylindrical space as the previously published result, and the improvement in energy storage characteristics are reported.

It is shown that the new problem formulation can lead to rotors with optimal volumetric energy density, which implicitly maintain high weight energy density and a good shape factor.

4.3.1 Example 1

With the two materials given in Table 4.1, the optimization problem is solved, while using the Tsai-Wu failure criterion and the proposed problem formulation. The

resulting optimal rotor is scaled to fit into the same cylindrical space as the rotor reported in Table 4.2 and it is constrained to have the same axial thickness. The effect of residual stress is not considered.

Details of the resulting optimal rotor are given in Table 4.7. When compared to the previous result, published by Krack *et al.* [1], the new problem formulation has lead to an increase in energy density for this problem. The volumetric energy density has improved by almost 8% while the weight energy density and shape factor have also improved.

4.3.2 Example 2

The use of the new problem formulation can also be demonstrated with the material set from Table 4.3. Here the same three failure criteria that were used by Arvin and Bakis [2] are applied simultaneously and a new optimal design is found. The effect of residual stress is included by specifying an initial temperature difference and shrink fits are not allowed. The resulting rotor is scaled to fit into the same cylindrical space as the previously reported result.

The details of the resulting optimal rotor are given in Table 4.7. For this example the volumetric energy density has improved by almost 15% while the weight energy density and shape factor have improved dramatically. These improvements are possible even without the use of shrink fits between layers. However, this new optimal rotor does not accommodate the same inner ring of steel and magnets as prescribed by Arvin and Bakis [2].

4.3.3 Example 3

For this example the material set from Table 4.5 is used together with the same failure criterion used by Ha [3]. The resulting design is scaled to have an outer radius of 155.22 mm and a height of 100 mm.

The details of the resulting optimal rotor are given in Table 4.7. When compared to Ha's result, the new problem formulation allows for a 75.5% improvement in the volumetric energy density, while effectively doubling the weight energy density and shape factor. Such dramatic improvement serves to demonstrate the necessity of using the proposed new problem formulation.

The stress distributions in each of the optimal rotors that are reported here, are plotted in Appendix A.

In summary, it has been shown in this chapter that significant improvements in the energy density of thick rotors may be possible if the optimization problems are formulated carefully. The respective improvements are shown in Figure 4.4. In the

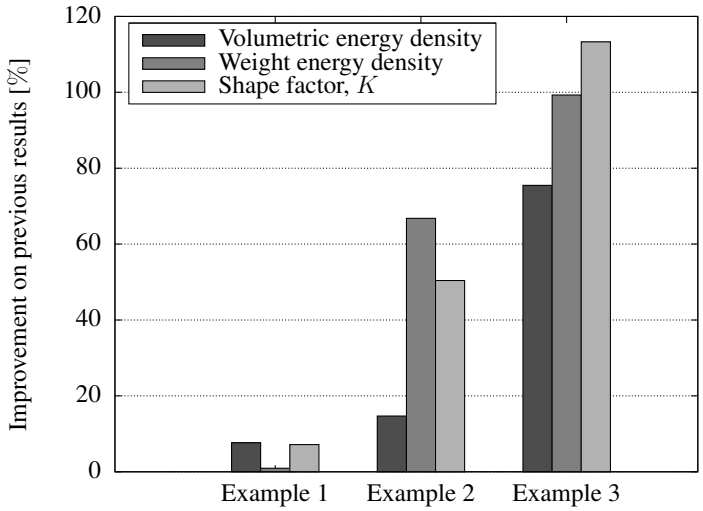


Figure 4.4: The improvement in energy storage parameters presented by the proposed optimal designs.

next chapter, the problem of selecting the best materials on which to base a design is discussed.

Chapter 5

How to choose the best set of materials

Finding the most appropriate material set on which to base a design optimization can be a daunting task. In this chapter the problem of material selection is discussed and some new ideas are put forward.

In the previous chapter it was shown how to formulate the optimization problem in order to find the optimal distribution of a set of materials through a rotor. By formulating the optimization problem carefully, rotor designs with very high energy density can be found, but only if the right materials are available. At present, the available literature gives little guidance on how these materials should be chosen from the wealth of different composites that are available.

The examples from the previous chapter demonstrate that authors typically take only a few materials into account in an analysis. This is probably due to the increased number of design variables that have to be considered and the associated increase in computational cost for solving higher dimensional problems. In this chapter it is shown how the problem of material selection can be tackled by solving a series of problems of low dimensionality whilst working with a large set of available materials. The approach is tested on an example material set, after which its use is demonstrated by finding the optimal rotor from a set of ten materials.

5.1 Finding optimal material set for k-dimensional problem

5.1.1 Material sequence

As mentioned in Section 2.1, it has been found in practice that the optimal rotor from a given set of materials is one for which the materials are arranged in order of increasing specific stiffness. This allows flywheel rotor designers to solve design problems far more easily than would have been the case if every possible material sequence had to be evaluated.

This approach has been adopted in literature [2, 1], and it was also employed when solving the design problems in Chapter 4. However, it remains unproven, and so it is formulated as a proposition rather than a fact:

Proposition 5.1 *Consider a set of cylindrical rotors made from concentric layers of a given set of materials. The rotor with greatest energy density will be one for which the materials are arranged in order of increasing specific stiffness along the radius.*

This proposition serves to make a very difficult combinatorial problem manageable, so it is definitely of practical use. However, even when only a single material sequence needs to be evaluated in order to solve a design optimization problem, a designer may still run into computational difficulty if the material set is not sufficiently small.

5.1.2 Material combination

If Proposition 5.1 is true, then in order to find an optimal rotor from a set of n available materials, it is necessary to solve one optimization problem, of at least n-dimensions. Solving such a problem is usually only feasible for small values of n. For larger values the problem quickly becomes too computationally intensive. This is reflected in literature, where designers typically only consider two [17, 19, 1], three [3, 2] or four [23] materials in an analysis.

A more practical approach might be to solve several smaller problems, and then infer the best combination of materials from the results of such analyses. However, this approach is also susceptible to computational difficulties: Suppose, for example, that we are interested in a rotor with a small number of materials, say, k, and we want the best k-material rotor from a set of n available materials. The number of k-dimensional problems that needs to be solved in order to analyze every possible unordered combination of materials is given by the binomial coefficient

$$\binom{n}{k} = \frac{n!}{(n-k)!k!}, \tag{5.1}$$

which represents a number that easily becomes unpractical if much more than a handful of materials are to be considered.

In reality, the set of available composite materials is very large indeed. Not only is there a multitude of different possible combinations of fibers and matrix materials, but subtle differences in fiber volume fraction, or manufacturing setup or even the curing parameters may influence material properties significantly. In the face of such a large set of available materials, and limited computational resources, a designer can do little but make an educated guess as to the right subset of material properties to consider for a design.

To help address this problem, we make the following proposition, which allows us to ignore some of the possible material combinations mentioned above:

Proposition 5.2 *Consider a large set, containing n different materials. Now let S_1 be a set containing only the material of the optimal single-material rotor. Also, let S_2 be the set containing only those materials constituting the optimal two-material rotor. Then, we propose that S_1 is a subset of S_2, i.e. $S_1 \subset S_2$ and in general, $S_{k-1} \subset S_k$ for all $k \leq n$.*

If this is true, then it serves to save a significant amount of computational effort, making it possible to find optimal rotors from relatively large sets of available materials.

5.1.3 Computational implications

To demonstrate the implications of Proposition 5.2, suppose that we are looking for the optimal 5-material rotor from a set of 10 available materials. Instead of solving each of the 252 possible 5-dimensional optimization problems to find an overall solution, we proceed as follows:

First we solve ten 1-dimensional problems to find the optimal single-material rotor. Now we know that, according to the proposition above, this material must also be present in the optimal 2-material rotor. So, to find the optimal 2-material rotor, we only have to solve nine 2-dimensional problems to find the material that goes best with the solution from the previous step. For the next steps we have to solve eight 3-dimensional problems, and then seven 4-dimensional problems until, finally, we solve six 5-dimensional problems in the last step.

Since a 5-dimensional problem takes significantly more computational effort than any problem of fewer dimensions, we saved 246 very time-consuming problems at the expense of a number of smaller problems that are solved rather quickly.

5.2 Example test case

To test the validity of Proposition 5.2 over a certain material design space it is necessary to solve an optimization problem for each possible unordered combination of those materials. The results of such an analysis can be grouped according to the number of materials that were considered for each optimization problem. The proposition is then tested by scrutinizing the materials used for the best rotor in each group. The material from the optimal single-material rotor should also be contained in the material set of the optimal two-material rotor. These two materials should also be contained in the material set of the optimal three-material rotor, and so on.

5.2.1 Method of analysis

Plane stress rotors, that are very thin in the axial direction, were deemed sufficient for the analysis. The problem formulation given in Chapter 4 was used together with the analytical methods described in Chapter 6. The finite element models were constructed in such a way that each rotor was made up from one hundred elements in the radial direction and a single element in the axial direction.

The quadratic Tsai-Wu failure criterion was used to assess failure, and residual stress from curing was not taken into account. The analysis did not allow any interference fits between layers.

5.2.2 Results for test case

The test was carried out for the set of five materials shown in Table 5.1. For each possible, unordered combination of materials from this set a design optimization problem was solved. The solutions are given in Table 5.2, where the results are grouped according to the number of materials that were considered for each design optimization problem. The optimal rotor for each group is highlighted.

Upon inspection it becomes apparent that Proposition 5.2 holds true for the set of five materials in Table 5.2. That is, there is no optimal rotor that does not contain all of the materials in the optimal subset of that rotor's materials. If we were to apply the proposition in searching for the overall optimal rotor from this set of materials, then we would arrive at the correct optimum, and we would have saved some computational effort.

It should be noted that the proposition also holds for each possible subset of materials that can be formulated from the given set of five. It has thus been validated over the whole design space spanned by the selected five materials.

Property	Description	m_1	m_2	m_3	m_4	m_5	Units
E_1	Young's modulus	55.0	80.0	230.0	38.6	130.0	[GPa]
E_2		16.0	5.5	6.6	8.27	9.0	[GPa]
G_{12}	shear modulus	7.6	2.1	4.8	4.14	4.55	[GPa]
ν_{12}	Poisson's ratio	0.26	0.31	0.25	0.26	0.3	
X	tensile strength	1800	2000	1100	1062	1800	[MPa]
X'	compressive strength	690	280	620	610	1400	[MPa]
Y	tensile strength	40	20	21	31	80	[MPa]
Y'	compressive strength	140	140	170	118	168	[MPa]
ρ	density	2000	1380	1630	1800	1600	[kg/m^3]
X/ρ	strength-to-density	250.0	402.6	187.5	163.9	312.5	[Wh/kg]

Table 5.1: The set of materials that were used for testing purposes. For the material strength parameters, X corresponds to the fiber direction, and Y to the direction normal to the fiber direction.

5.3 Application to larger set of materials

Once our methods of analysis can handle large material sets, it becomes possible to test the approach that we have taken for trying to improve the energy density of rotors. In the previous chapters it was shown how energy density can be improved by layering several materials into the same rotor, but what is the maximum improvement that we can expect when using this approach?

A step may be taken towards answering this question by analysing a large material set and reporting the increase in energy density that is made possible by increasing the number of materials. In this section, the approach is demonstrated for a set of ten materials. The same five materials from the previous section were used, together with the five materials listed in Table 5.3.

The proposition that was introduced in the previous section was then coded into a Python script to implement the material selection logic. First the optimal single-material rotor was found and then additional layers were added sequentially until no further improvements were possible. The results from the analysis are shown in Table 5.4, where the optimal rotor from each group is highlighted.

From the results of the analysis, it can be seen that the best two-material rotor presents an improvement of roughly 33% on the energy density of the optimal single-material counterpart. By adding a third material, an additional 12% improvement is made possible. The overall optimal rotor was found to consist of only three materials, of which two are made from high strength carbon fibers and one from s-glass.

The shape factor for the overall optimal rotor was found to be $K = 0.44$, which is quite close to the theoretical maximum of 0.5. This suggests that the rotor is relatively close to fully stressed at the point of failure. With the current set of

Material Set	E_{sp} [Wh/m^3]	$\frac{r_i}{r_o}$	$t_1, t_2, etc.$
m_1	96093	0.74	1.0
m_2	107183	0.738	1.0
m_3	49080	0.728	1.0
m_4	55154	0.73	1.0
m_5	**118311**	**0.668**	**1.0**
m_1, m_2	152021	0.581	0.434, 0.566
m_1, m_3	96093	0.74	1.0, 0.0
m_4, m_1	96093	0.74	0.0, 1.0
m_1, m_5	**152243**	**0.466**	**0.436, 0.564**
m_2, m_3	107183	0.739	1.0, 0.0
m_4, m_2	127554	0.595	0.244, 0.756
m_2, m_5	129272	0.592	0.436, 0.564
m_4, m_3	79116	0.357	0.588, 0.412
m_5, m_3	118311	0.668	1.0, 0.0
m_4, m_5	144050	0.428	0.39, 0.61
m_1, m_2, m_3	152021	0.581	0.435, 0.565, 0.0
m_4, m_1, m_2	152022	0.581	0.0, 0.435, 0.565
m_1, m_2, m_5	152243	0.466	0.436, 0.0, 0.564
m_4, m_1, m_3	96093	0.74	0.0, 1.0, 0.0
m_1, m_5, m_3	152244	0.466	0.436, 0.564, 0.0
m_4, m_1, m_5	**153580**	**0.441**	**0.104, 0.358, 0.538**
m_4, m_2, m_3	127554	0.595	0.244, 0.756, 0.0
m_2, m_5, m_3	129272	0.592	0.436, 0.564, 0.0
m_4, m_2, m_5	147722	0.452	0.365, 0.127, 0.508
m_4, m_5, m_3	144050	0.428	0.39, 0.61, 0.0
m_4, m_1, m_2, m_3	152022	0.581	0.0, 0.435, 0.565, 0.0
m_1, m_2, m_5, m_3	152244	0.466	0.436, 0.0, 0.564, 0.0
m_4, m_1, m_2, m_5	**153580**	**0.441**	**0.104, 0.358, 0.0, 0.538**
m_4, m_1, m_5, m_3	**153580**	**0.441**	**0.104, 0.358, 0.538, 0.0**
m_4, m_2, m_5, m_3	147722	0.452	0.365, 0.127, 0.508, 0.0
m_4, m_1, m_2, m_5, m_3	**153580**	**0.441**	**0.104, 0.358, 0.0, 0.538, 0.0**

Table 5.2: The optimal rotor for each possible unordered subset from the set of five materials m_1, m_2, m_3, m_4 and m_5. The materials are arranged in order of increasing specific stiffness.

Property	Description	m_6	m_7	m_8	m_9	m_10	Units
E_1	Young's modulus	155.0	148.0	195.0	278.0	138.0	[GPa]
E_2		9.0	7.8	7.5	6.9	9.0	[GPa]
G_{12}	shear modulus	4.55	3.9	3.75	3.45	3.46	[GPa]
ν_{12}	Poisson's ratio	0.3	0.34	0.3	0.3	0.3	
X	tensile strength	2900	1450	1800	1280	2940	[MPa]
X'	compressive strength	1600	928	928	579	1600	[MPa]
Y	tensile strength	70	50	50	50	25	[MPa]
Y'	compressive strength	168	70	70	70	168	[MPa]
ρ	density	1600	1570	1570	1590	1600	[kg/m^3]
X/ρ	strength-to-density	503.5	256.5	318.5	223.6	510.4	[Wh/kg]

Table 5.3: The set of additional materials for evaluating the use of Proposition 5.2. For the material strength parameters, X corresponds to the fiber direction, and Y to the direction normal to the fiber direction.

materials, further improvements may be possible by making use of other methods such as the inclusion of pre-stress by shrink fitting for example. However, since the optimal rotor presented here already has a relatively high shape factor, the additional cost associated with such methods may outweigh the possible benefits.

It may be worth considering even larger material sets during a further study. Also, the development of a formal proof (or disproval) of Propositions 5.1 and 5.2 would be of great interest.

Material Set	E_{sp} [Wh/m^3]	$\frac{r_i}{r_o}$	$t_1, t_2, etc.$
m_1	96093	0.74	1.0
m_2	107183	0.738	1.0
m_3	49080	0.728	1.0
m_4	55154	0.73	1.0
m_5	118311	0.668	1.0
m_6	**159174**	**0.736**	**1.0**
m_7	98200	0.673	1.0
m_8	122827	0.682	1.0
m_9	135945	0.031	1.0
m_10	87804	0.84	1.0
m_1, m_6	**212538**	**0.467**	**0.429, 0.571**
m_2, m_6	192310	0.623	0.468, 0.532
m_6, m_3	159174	0.736	1.0, 0.0
m_4, m_6	191399	0.439	0.388, 0.612
m_5, m_6	159174	0.736	0.0, 1.0
m_7, m_6	159174	0.736	0.0, 1.0
m_6, m_8	159174	0.736	1.0, 0.0
m_6, m_9	136663	0.1	0.494, 0.506
m_10, m_6	161154	0.73	0.074, 0.926
m_1, m_2, m_6	223978	0.494	0.353, 0.196, 0.451
m_1, m_6, m_3	212540	0.467	0.429, 0.571, 0.0
m_4, m_1, m_6	212540	0.467	0.0, 0.429, 0.571
m_1, m_5, m_6	212540	0.467	0.429, 0.0, 0.571
m_1, m_7, m_6	212539	0.467	0.429, 0.0, 0.571
m_1, m_6, m_8	212540	0.467	0.429, 0.571, 0.0
m_1, m_6, m_9	212540	0.467	0.429, 0.571, 0.0
m_1, m_10, m_6	**231403**	**0.499**	**0.436, 0.102, 0.462**
m_1, m_2, m_10, m_6	231409	0.499	0.436, 0.0, 0.102, 0.462
m_1, m_10, m_6, m_3	231409	0.499	0.436, 0.102, 0.463, 0.0
m_4, m_1, m_10, m_6	231409	0.499	0.0, 0.436, 0.102, 0.462
m_1, m_5, m_10, m_6	231409	0.499	0.436, 0.0, 0.102, 0.462
m_1, m_10, m_7, m_6	231409	0.499	0.436, 0.102, 0.0, 0.462
m_1, m_10, m_6, m_8	231409	0.499	0.436, 0.102, 0.463, 0.0
m_1, m_10, m_6, m_9	231408	0.499	0.436, 0.102, 0.463, 0.0

Table 5.4: Results from a design optimization where Proposition 5.2 is applied to a set of ten materials.

Chapter 6

Stress analysis and optimization

In this chapter the methods of analysis that were used in this study are presented. These include the axisymmetric finite element method, used with some relevant failure criteria as well as the particle swarm optimization algorithm and a Bayesian stopping criterion.

The analytical methods that were used for solving the design problems encountered in this study, are outlined in the following sections. These were implemented from scratch as a Python program, and fragments of the code can be found in Appendix C.

The program can roughly be divided into three layers of code, which have each been afforded a section in this chapter. At the start of each section the program flow is demonstrated in a flow-chart, whereafter the relevant theory on which it was based is discussed in detail.

6.1 Stress and strength analysis

For this study the axisymmetric finite element method is used for stress analysis, together with two popular failure criteria for analyzing strength. The program flow of the corresponding code can be seen in Figure 6.1.

Other methods are available for stress analysis. Examples include analytical methods such as those proposed by Perez-Aparicio [18] or Ha [3]. Such methods can be seen in action in the work of Arvin [2] and Krack [1] and many of Ha's publications. In certain cases they have been demonstrated to be sufficiently accurate, but they do rely on relatively simple one-dimensional models. These models inevitably involve some assumptions regarding the strain state in directions that are

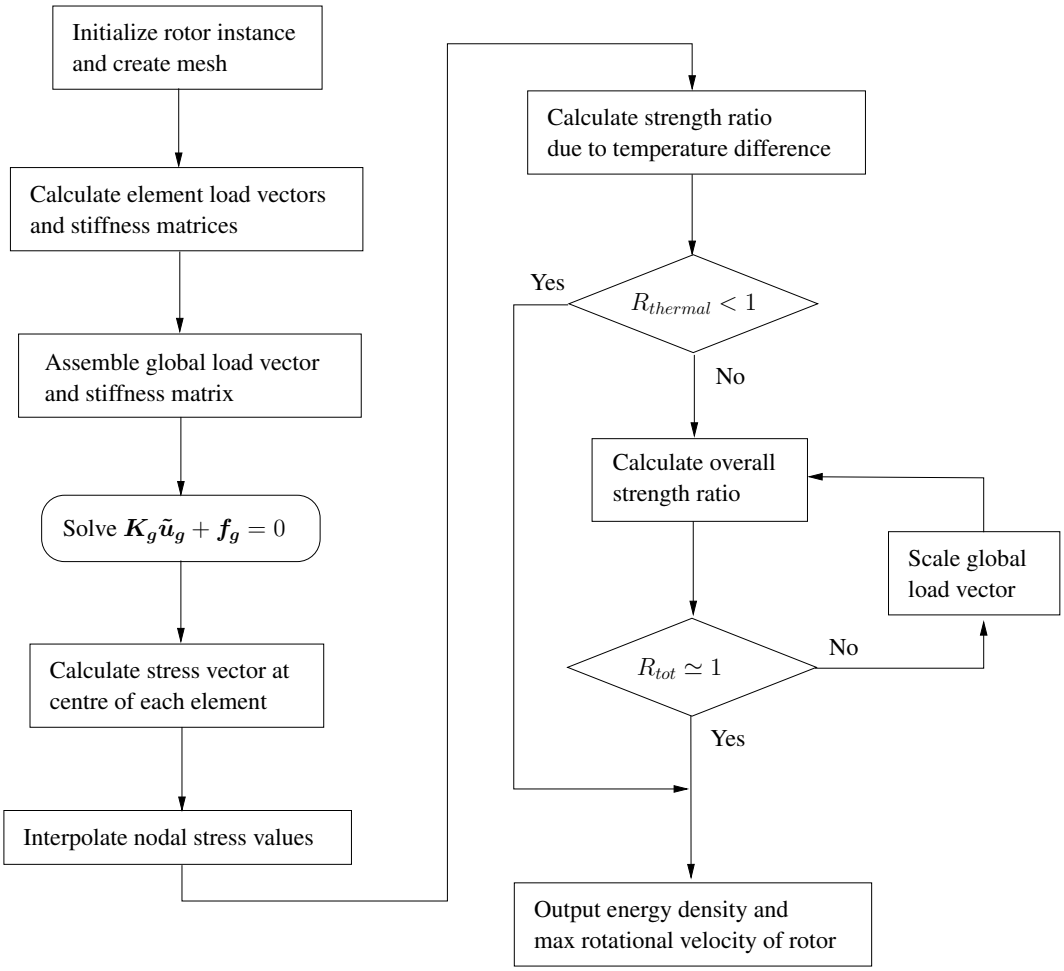


Figure 6.1: Overview of program flow for stress and strength analysis routine.

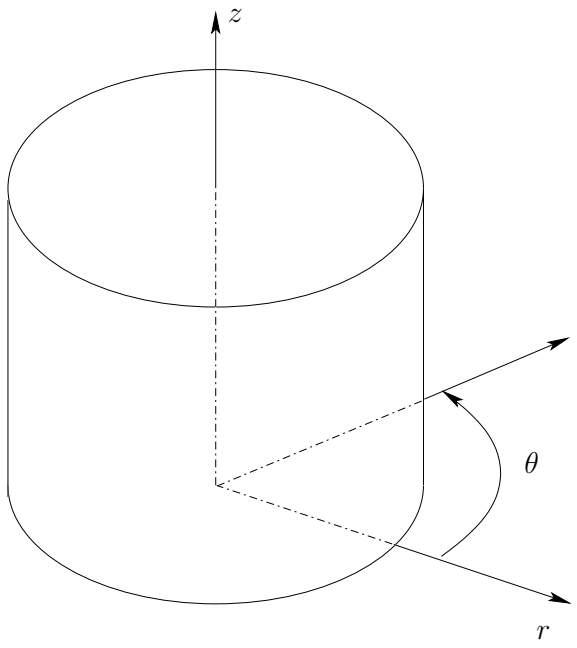


Figure 6.2: The cylindrical coordinate system that is used for the derivations in this chapter.

not directly captured by the model, and these may be a source of inaccuracy if not handled carefully.

The axisymmetric finite element method makes use of a two-dimensional model to represent a rotating body, and for this study it represents a robust method of stress analysis. Other authors that have also opted to use this approach include Genta [8].

The contents of this section are based on the outlay given in the classic text by Zienkiewicz *et al.* [24], but texts by Chandrupatla [25] and Cook [26] were also consulted. Figure 6.2 shows the cylindrical coordinate system that is used in the derivations in this section.

6.1.1 Governing equations

The elastic behavior of rotating cylinders is well described by a two-dimensional, axisymmetric model. However, one important implication of such a model should be kept in mind, namely that unsymmetrical modes of vibration cannot be represented. For this study, natural frequencies and modes of vibration were not important, but for other applications where they are important it would be necessary to make use of a three-dimensional model to capture these modes.

Displacement functions

The main assumption on which the axisymmetric model is based, is that the displacement field $\boldsymbol{u}$ varies only in two dimensions, such that

$$\boldsymbol{u} = \begin{Bmatrix} u(r,z) \\ v(r,z) \end{Bmatrix}, \tag{6.1}$$

where u is the radial displacement and v the axial displacement.

Strain-displacement relationship

If the displacement field is known, then strains can be calculated as

$$\boldsymbol{\varepsilon} = \begin{Bmatrix} \varepsilon_r \\ \varepsilon_z \\ \varepsilon_\theta \\ \gamma_{rz} \end{Bmatrix} = \mathcal{S}\boldsymbol{u}, \tag{6.2}$$

where the subscripts follow from Figure 6.2, and $\mathcal{S}$ is the differential operator

$$\mathcal{S} = \begin{bmatrix} \frac{\partial}{\partial r} & 0 \\ 0 & \frac{\partial}{\partial z} \\ \frac{1}{r} & 0 \\ \frac{\partial}{\partial z} & \frac{\partial}{\partial r} \end{bmatrix}. \tag{6.3}$$

Equilibrium and boundary conditions

For an axisymmetric body, the equilibrium equations state that internal strain energy caused by loading has to be equal to the work done by the applied loads, such that

$$\boldsymbol{\varepsilon}^{\boldsymbol{T}}\boldsymbol{\sigma} - \boldsymbol{u}^{\boldsymbol{T}}\boldsymbol{b} = 0. \tag{6.4}$$

By substituting (6.2) into (6.4), equilibrium becomes

$$\mathcal{S}^T\boldsymbol{\sigma} - \boldsymbol{b} = \boldsymbol{0}, \tag{6.5}$$

where $\boldsymbol{\sigma}$ is the stress vector

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_r \\ \sigma_z \\ \sigma_\theta \\ \tau_{rz} \end{bmatrix}, \tag{6.6}$$

and $\boldsymbol{b}$ is the body force vector

$$\boldsymbol{b} = \begin{bmatrix} b_r \\ b_z \end{bmatrix}. \tag{6.7}$$

Here, only body loads are considered, such as those caused by acceleration or gravity. Depending on the application, it may be necessary to include surface tractions in (6.4) and (6.5) above.

The displacement boundary conditions are specified on each point of the boundary

$$\boldsymbol{u} = \bar{\boldsymbol{u}} \tag{6.8}$$

where $\bar{\boldsymbol{u}}$ are prescribed values.

Compliance and elasticity

The constitutive relation for a solid material can be written as

$$\boldsymbol{\sigma} = \boldsymbol{D}(\varepsilon - \varepsilon_{\mathbf{0}}) + \boldsymbol{\sigma_0}, \tag{6.9}$$

where $\boldsymbol{\varepsilon_0}$ represents initial strain, and $\boldsymbol{\sigma_0}$ initial stress. Here $\boldsymbol{D}$ is known as the elasticity matrix.

Alternatively, the constitutive relation is also given by

$$\varepsilon = \boldsymbol{D^{-1}}(\boldsymbol{\sigma} - \boldsymbol{\sigma_0}) + \boldsymbol{\varepsilon_0}, \tag{6.10}$$

where $\boldsymbol{D^{-1}}$ is known as the compliance matrix.

For the circumferentially wound composite fiber materials that are considered in this study, the compliance matrix can be shown to be

$$\boldsymbol{D^{-1}} = \begin{bmatrix} \frac{1}{E_r} & -\frac{\nu_{rz}}{E_z} & -\frac{\nu_{r\theta}}{E_\theta} & 0 \\ -\frac{\nu_{rz}}{E_z} & \frac{1}{E_z} & -\frac{\nu_{z\theta}}{E_r} & 0 \\ -\frac{\nu_{r\theta}}{E_\theta} & -\frac{\nu_{z\theta}}{E_r} & \frac{1}{E_\theta} & 0 \\ 0 & 0 & 0 & \frac{1}{G_{rz}} \end{bmatrix}, \tag{6.11}$$

where E denotes the modulus of elasticity, G the shear modulus and ν the Poisson's ratio of a material.

Initial strain

For this study, the effects of initial strain due to the curing process was taken into account for some of the problems in Chapter 4. This was accomplished by including

a temperature difference as part of the initial strain term in (6.9) and (6.10) such that

$$\boldsymbol{\varepsilon_0} = \Delta T \begin{Bmatrix} \alpha_r \\ \alpha_z \\ \alpha_\theta \\ 0 \end{Bmatrix}, \tag{6.12}$$

where ΔT is a temperature difference and the α-constants are the coefficients of thermal expansion of the material. Here, the temperature difference is taken to be the difference between the curing temperature of a rotor and its operating temperature.

6.1.2 Finite element approximation

In order to solve the equations from the section above, some approximations are incorporated. Firstly, the equilibrium of (6.4) is only applied in a weak sense by introducing a small variation while requiring that the essential boundary conditions should still be satisfied. Equilibrium can then be expressed as

$$\int_\Omega \delta\boldsymbol{\varepsilon}^T \boldsymbol{\sigma}\, d\Omega - \int_\Omega \delta\boldsymbol{u}^T \boldsymbol{b}\, d\Omega = 0, \tag{6.13}$$

where Ω is the volume of the body. By applying (6.9), this can also be written as

$$\int_\Omega \delta\boldsymbol{\varepsilon}^T [\boldsymbol{\sigma_0} + \boldsymbol{D}(\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon_0})]\, d\Omega - \int_\Omega \delta\boldsymbol{u}^T \boldsymbol{b}\, d\Omega = 0. \tag{6.14}$$

In other words, equilibrium only holds true in an average sense over the body, but not at every point within the body.

The body is then divided into a mesh of distinct elements to allow for element-wise integration over complex geometries. An example of a meshed rotor model is shown in Figure 6.3. The equilibrium equation now becomes

$$\sum_e \int_{\Omega^e} \delta\boldsymbol{\varepsilon}^T [\boldsymbol{\sigma}_0 + \boldsymbol{D}(\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}_0)]\, d\Omega - \sum_e \int_{\Omega^e} \delta\boldsymbol{u}^T \boldsymbol{b}\, d\Omega = 0, \tag{6.15}$$

where the subscript e denotes that integrations are taken element-wise and that the results are then summated over all elements within the body.

The next approximation is to interpolate the displacement field from nodal values by making use of shape functions. These serve to simplify the element-wise integrations in (6.15). The displacement field within an element now becomes

$$\boldsymbol{u} \approx \hat{\boldsymbol{u}} = \begin{Bmatrix} \hat{u} \\ \hat{v} \end{Bmatrix} = \sum N_a \begin{Bmatrix} \tilde{u}_a \\ \tilde{v}_a \end{Bmatrix} = \sum N_a \tilde{\boldsymbol{u}}_{\boldsymbol{a}}, \tag{6.16}$$

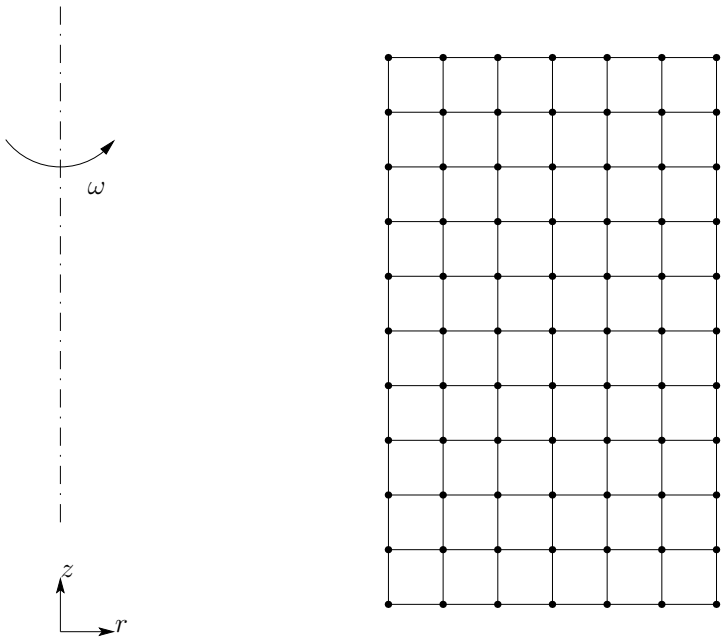


Figure 6.3: An example of a meshed rotor model with 77 nodes and 154 degrees of freedom.

where N denotes a shape function, and $\tilde{u}$ and $\tilde{v}$ are nodal displacements. The subscript a denotes a node within the element, and the summation is carried out over all the nodes, each with their respective shape function.

It should be noted that the shape functions do not necessarily need to represent the anticipated displacement field exactly. In fact, for an arbitrary displacement field, a model may still converge in the limit of mesh refinement. In Appendix B the patch test is applied to show that the four-noded bi-linear elements that were used for this study satisfy the requirements for convergence.

The strains at any point within an element can now be calculated from

$$\boldsymbol{\varepsilon} = \begin{Bmatrix} \varepsilon_r \\ \varepsilon_z \\ \varepsilon_\theta \\ \gamma_{rz} \end{Bmatrix} = \sum_a \begin{bmatrix} \frac{\partial N_a}{\partial r} & 0 \\ 0 & \frac{\partial N_a}{\partial z} \\ \frac{N_a}{r} & 0 \\ \frac{\partial N_a}{\partial z} & \frac{\partial N_a}{\partial r} \end{bmatrix} \begin{Bmatrix} \tilde{u}_a \\ \tilde{v}_a \end{Bmatrix} = \sum_a \boldsymbol{B_a} \tilde{\boldsymbol{u}}_{\boldsymbol{a}}, \tag{6.17}$$

where $\boldsymbol{B}$ is called the strain-displacement matrix.

By substituting (6.16) and (6.17) into (6.15), the weak form of equilibrium can be

written as

$$\sum_e \int_{\Omega^e} \delta\tilde{\boldsymbol{u}}_{\boldsymbol{a}}^{\boldsymbol{T}} \boldsymbol{B}_{\boldsymbol{a}}^{\boldsymbol{T}} [\boldsymbol{\sigma_0} + \boldsymbol{D}(\boldsymbol{B}_{\boldsymbol{a}}\tilde{\boldsymbol{u}}_{\boldsymbol{a}} - \boldsymbol{\varepsilon_0})]\, d\Omega - \sum_e \int_{\Omega^e} \delta\tilde{\boldsymbol{u}}_{\boldsymbol{a}}^{\boldsymbol{T}} N_a \boldsymbol{b}\, d\Omega = 0, \quad (6.18)$$

or,

$$\sum_e \delta\tilde{\boldsymbol{u}}_{\boldsymbol{a}}^{\boldsymbol{T}} \left[\int_{\Omega^e} \boldsymbol{B}_{\boldsymbol{a}}^{\boldsymbol{T}} [\boldsymbol{\sigma_0} + \boldsymbol{D}(\boldsymbol{B}_{\boldsymbol{a}}\tilde{\boldsymbol{u}}_{\boldsymbol{a}} - \boldsymbol{\varepsilon_0})]\, d\Omega - \int_{\Omega^e} N_a \boldsymbol{b}\, d\Omega \right] = 0. \quad (6.19)$$

Now, summing the element integrals and noting that the small variation is arbitrary, we get a system of linear equations

$$\boldsymbol{K}_{\boldsymbol{a}}\tilde{\boldsymbol{u}}_{\boldsymbol{a}} + \boldsymbol{f}_{\boldsymbol{a}} = \boldsymbol{0}, \quad (6.20)$$

where

$$\boldsymbol{K}_{\boldsymbol{a}} = \sum_e \int_{\Omega^e} \boldsymbol{B}_{\boldsymbol{a}}^{\boldsymbol{T}} \boldsymbol{D} \boldsymbol{B}_{\boldsymbol{a}}\, d\Omega, \quad (6.21)$$

$$\boldsymbol{f}_{\boldsymbol{a}} = \sum_e \int_{\Omega^e} [\boldsymbol{B}_{\boldsymbol{a}}^{\boldsymbol{T}} (\boldsymbol{\sigma_0} - \boldsymbol{D}\boldsymbol{\varepsilon_0}) - N_a \boldsymbol{b}]\, d\Omega, \quad (6.22)$$

Here $\boldsymbol{K}_{\boldsymbol{a}}$ denotes the element stiffness matrix and $\boldsymbol{f}_{\boldsymbol{a}}$ denotes the element load vector.

An advantage of using shape functions to interpolate displacements is that Gauss-quadrature can be used to perform the integrations in (6.21) and (6.22). If the order of the shape functions are known, then a quadrature rule can be selected which integrates these functions exactly.

The quadrature rule prescribes the coordinates that are to be sampled in addition to a weighting parameter, W to be assigned to each sampling. The coordinates are best defined in terms of an element's local coordinate system, consisting of the two axes ξ and η. The subscript l is used to denote a quadrature point at (ξ_l, η_l) with a corresponding weight of W_l.

An additional scalar, J, called the Jacobian is introduced at this point to perform mapping from the element's local coordinate system to the global coordinates. The Jacobian is defined as the determinate of the Jacobian matrix

$$J = \text{detJ}, \quad (6.23)$$

where the Jacobian matrix is given by

$$\text{J} = \begin{bmatrix} \frac{\partial u}{\partial \xi} & \frac{\partial v}{\partial \xi} \\ \frac{\partial u}{\partial \eta} & \frac{\partial v}{\partial \eta} \end{bmatrix}. \quad (6.24)$$

Equations (6.21) and (6.22) can now be rewritten as follows:

$$\begin{aligned} \boldsymbol{K}_{\boldsymbol{a}}^{e} &= \int_{-1}^{1}\int_{-1}^{1} \boldsymbol{B}_{\boldsymbol{a}}^{\boldsymbol{T}}(\xi,\eta)\boldsymbol{D}\boldsymbol{B}_{\boldsymbol{a}}(\xi,\eta)J(\xi,\eta)\,d\xi\,d\eta \\ &\approx \sum_{l} \boldsymbol{B}_{\boldsymbol{a}}^{\boldsymbol{T}}(\xi_l,\eta_l)\boldsymbol{D}\boldsymbol{B}_{\boldsymbol{a}}(\xi_l,\eta_l)J(\xi_l,\eta_l)W_l \end{aligned} \tag{6.25}$$

$$\begin{aligned} \boldsymbol{f}_{\boldsymbol{a}}^{e} &= \int_{-1}^{1}\int_{-1}^{1} [\boldsymbol{B}_{\boldsymbol{a}}^{\boldsymbol{T}}(\xi,\eta)(\boldsymbol{\sigma_0} - \boldsymbol{D}\boldsymbol{\varepsilon_0}) - N_a(\xi,\eta)\boldsymbol{b}]J(\xi,\eta)\,d\xi\,d\eta \\ &\approx \sum_{l} [\boldsymbol{B}_{\boldsymbol{a}}^{\boldsymbol{T}}(\xi_l,\eta_l)(\boldsymbol{\sigma_0} - \boldsymbol{D}\boldsymbol{\varepsilon_0}) - N_a(\xi_l,\eta_l)\boldsymbol{b}]J(\xi_l,\eta_l)W_l \end{aligned} \tag{6.26}$$

In our implementation, a second-order Gauss rule was applied, for which the sampling points (ξ_l, η_l) are $(1/\sqrt{3}, 1/\sqrt{3})$, $(-1/\sqrt{3}, 1/\sqrt{3})$, $(-1/\sqrt{3}, -1/\sqrt{3})$, and $(1/\sqrt{3}, -1/\sqrt{3})$, while the weighting parameter $W_l = 1$ for each of the points.

Once the elemental load vector and stiffness matrix has been calculated, the final steps of the finite element method is to assemble all of the available information into a global load vector $\boldsymbol{f_g}$ and stiffness matrix $\boldsymbol{K_g}$. By solving a system of linear equations, the nodal displacements, $\tilde{\boldsymbol{u}}_{\boldsymbol{g}}$ can be found according to

$$\boldsymbol{K_g}\tilde{\boldsymbol{u}}_{\boldsymbol{g}} + \boldsymbol{f_g} = \boldsymbol{0}. \tag{6.27}$$

Once the nodal displacements are known it is a simple matter to find the strain state in an element from (6.17) and the stress state from the constitutive relation.

The Python implementation of the axisymmetric, four-noded, bilinear element that was used for this study is given in Appendix C.

6.1.3 Strength analysis

Once the stress state of a model is known, the next task is to assess whether the stress state constitutes failure or not. To analyze the strength of the axisymmetric finite element models, two failure criteria were implemented, namely the *Maximum Stress*- and the quadratic *Tsai-Wu* criteria. This is consistent with the approaches taken by some of the authors cited in this work.

Maximum stress

The first criterion simply states that the magnitude of each stress component at every point within a model should be below the applicable material strength, as

determined in a simple tensile-, compressive- or shear test. If this is not the case, then failure has occurred. This criterion can be expressed as

$$-X' < \sigma_1 < X \tag{6.28}$$

for stress along the fiber direction, and

$$-Y' < \sigma_2 < Y \tag{6.29}$$

normal to the fiber direction, while the shear component of stress must obey

$$|\tau_{12}| < S. \tag{6.30}$$

The safety factor, or strength ratio, R, is then given by

$$R = \min\left(\frac{X}{\sigma_1}, \frac{Y}{\sigma_2}, \frac{S}{\tau_{12}}\right) \tag{6.31}$$

for the case of tensile loading, or a very similar expression for compressive loading.

Quadratic Tsai-Wu

The quadratic Tsai-Wu criterion [27] is often used for failure analysis of composites. It can be expressed as

$$\mathcal{FI} = F_1\sigma_1 + F_2\sigma_2 + F_{11}\sigma_1^2 + F_{22}\sigma_2^2 + 2F_{12}\sigma_1\sigma_2 + F_{66}\sigma_6^2 < 1, \tag{6.32}$$

where $\mathcal{FI}$ is the failure index and the various constants are defined as

$$F_1 = \frac{1}{X} - \frac{1}{X'}, \qquad F_2 = \frac{1}{Y} - \frac{1}{Y'},$$
$$F_{11} = \frac{1}{XX'}, \qquad F_{22} = \frac{1}{YY'},$$
$$F_{66} = \frac{1}{S},$$

and

$$F_{12} = F_{12}^*(F_{11}F_{22})^{0.5}.$$

Here F_{12}^* is chosen between -0.5 and 0.

At this point it may be tempting to use the failure index directly for purposes of optimization. However, it has been shown [22] that this may lead to unwanted results if the models are not evaluated close to the point of failure. Rather, it is

helpful to calculate the strength ratio, for which a constant matrix $\tilde{\boldsymbol{F}}$ can be defined as

$$\tilde{\boldsymbol{F}} = \begin{bmatrix} \frac{1}{XX'} & F_{12}^{*}\sqrt{\frac{1}{XX'}\frac{1}{YY'}} & F_{12}^{*}\sqrt{\frac{1}{XX'}\frac{1}{YY'}} & 0 \\ & \frac{1}{YY'} & F_{12}^{*}\frac{1}{YY'} & 0 \\ & & \frac{1}{YY'} & 0 \\ \text{symmetric} & & & \frac{1}{S^2} \end{bmatrix}, \tag{6.33}$$

and a constant vector $\bar{\boldsymbol{F}}$ can be defined as

$$\bar{\boldsymbol{F}} = \begin{bmatrix} \frac{1}{X} - \frac{1}{X'} & \frac{1}{Y} - \frac{1}{Y'} & \frac{1}{Y} - \frac{1}{Y'} & 0 \end{bmatrix}. \tag{6.34}$$

The strength ratio R can then be calculated from

$$R = \frac{\bar{\boldsymbol{F}}\boldsymbol{\sigma} + \sqrt{\left(\bar{\boldsymbol{F}}\boldsymbol{\sigma}\right)^2 + 4\boldsymbol{\sigma}\tilde{\boldsymbol{F}}\boldsymbol{\sigma}}}{2}, \tag{6.35}$$

where a value less than unity indicates failure.

6.2 Optimization

The class of design optimization problems encountered in this study are constrained, and often multi-modal problems, which are discontinuous in the first derivative. Many different optimization techniques exist for the solution of such problems, with some examples being the popular stochastic methods such as the *genetic algorithm*, *simulated annealing* or *differential evolution*. However, gradient-based algorithms may also be used if provision is made for handling the discontinuities in the first derivatives.

In this study we use the *particle swarm optimization algorithm* (PSOA), which is relatively simple to implement and performs reasonably well over a wide range of problems. It is also a stochastic method in the sense that it makes use of random variables to influence the choice of points to be sampled. Together with the other stochastic methods mentioned above, the PSOA is a zero-order method, in that it considers only the value, and not the gradient, of the objective function when a point is sampled. The gradient is often very expensive to compute.

The performance of the PSOA can vary greatly depending on the values that are chosen for the various optimization parameters. For this study the parameters were often tweaked to suit the particular problem at hand, allowing for reasonable computational efficiency to be achieved.

The program flow of the code that implemented this optimization routine can be seen in Figure 6.4. The description of the PSOA in this section is based on that given by Wood [28].

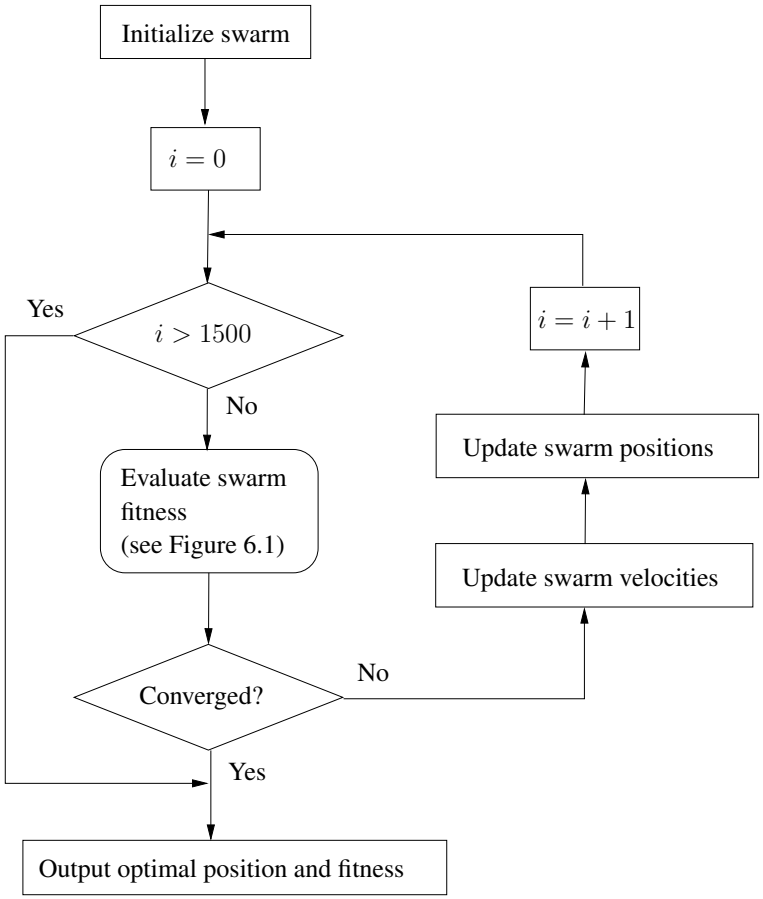


Figure 6.4: Overview of program flow for particle swarm optimization routine.

6.2.1 Overview of particle swarm optimization algorithm

As the name suggests, the particle swarm optimization algorithm tries to mimic the behavior of large groups of living organisms. What makes these groups special is that the individual members of the group may each act on its own will, while also taking in information from other members of the group. In a certain sense this exchange of information can make the group collectively more intelligent than any of the individual members are on their own. In the language of the PSOA a group is called a swarm, and a group member is called a particle.

Each particle of a swarm is characterized by its current position $\boldsymbol{x}$ and velocity $\boldsymbol{v}$ in the design space. Individually, each particle remembers its previous best position $\boldsymbol{p^i}$, and collectively the group remembers the best position $\boldsymbol{p^g}$ found by any member of the swarm.

The movement of the particle swarm through the design space is governed by two equations, namely the position update rule

$$\boldsymbol{x}^i_{n+1} = \boldsymbol{x}^i_n + \boldsymbol{v}^i_{n+1} \tag{6.36}$$

and the velocity update rule

$$\boldsymbol{v}^i_{n+1} = \boldsymbol{v}^i_n + c_1\boldsymbol{r}_1(\boldsymbol{p}^i_n - \boldsymbol{x}^i_n) + c_2\boldsymbol{r}_2(\boldsymbol{p}^g_n - \boldsymbol{x}^i_n) \tag{6.37}$$

where n denotes a time step and c_1 and c_2 are constants respectively known as the *cognitive-* and *social* scaling parameters. Some unpredictability is introduced into the equation by $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$, which are vectors containing random numbers between zero and one.

In the original version of (6.37), as suggested by Kennedy and Eberhart [29], it is not explicitly stated whether $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ should be random constants or random vectors, and so there exists some differences in the interpretation of the velocity update rule. However, it has been shown recently [30] that the use of random constants may have deleterious effects on the diversity of the swarm's velocity vectors after many iterations. Conversely, the use of random vectors may cause a loss of objectivity of the algorithm, and in cases where this is a serious problem it may be better to use random rotations within the velocity update rule [31].

6.2.2 Additional heuristics

By applying the position- and velocity update rules iteratively, the swarm moves through the design space in an unpredictable, but not entirely senseless fashion. By inspecting Equations (6.36) and (6.37) it can be seen that the PSOA is really a very simple algorithm to understand and to implement. However, the performance of the PSOA can be improved significantly for a wide range of problems by applying additional heuristics to the algorithm. One of the most popular of these is *dynamic*

inertia reduction, for which (6.37) is modified by adding a ω-term for the inertia of a particle as follows:

$$\boldsymbol{v}_{n+1}^{i} = \omega_n \boldsymbol{v}_n^i + c_1 \boldsymbol{r}_1(\boldsymbol{p}_n^i - \boldsymbol{x}_n^i) + c_2 \boldsymbol{r}_2(\boldsymbol{p}_n^g - \boldsymbol{x}_n^i). \tag{6.38}$$

Here the inertia changes at certain time steps, depending on the behavior of the swarm. If no improvement has been found in a specified number of steps, then the inertia is updated as

$$\omega_{n+1} = \alpha \omega_n, \tag{6.39}$$

where α is a constant typically chosen to be close to, but smaller than unity, say 0.95.

Another heuristic that is commonly implemented is to place a limit on the magnitude of the velocity vectors. If the velocity is constrained in magnitude to some maximum value v^{max}, then it is convenient to reduce this value concurrently with the inertia reduction by multiplying it with a control parameter β so that

$$\boldsymbol{v}_{n+1}^{max} = \beta \boldsymbol{v}_n^{max}, \tag{6.40}$$

where the value of β is typically also around 0.95.

6.2.3 Handling boundaries

The PSOA is perhaps best suited for unconstrained minimization, but it may also be adapted to handle constrained problems. For this study, constraints were only added indirectly by penalizing the fitness of particles that stray out of the feasible design region.

6.2.4 Criteria for convergence

There are several different approaches that may be used in order to determine whether a swarm has converged sufficiently. Some possibilities include a check on the value of the inertia parameter ω or on the value of the maximum particle velocity within the swarm. Other possibilities include a check on either the relative- or absolute improvements in the objective function value between swarm iterations.

For this study, convergence was evaluated by a combination of two checks: the first prescribed that for at least the ten most recent iterations there should not have been any improvement in the objective function value. Here, a small tolerance of 0.05% was allowed to account for numerical noise. The second requirement was that all particles within the swarm should be sufficiently close to the best known position, such that

$$\sqrt{\boldsymbol{\delta}_i^T \boldsymbol{\delta}_i} < 0.005, \quad \text{for each } \; i = 1, 2, \cdots, p. \tag{6.41}$$

Here, p denotes the number of particles within the swarm, and

$$\boldsymbol{\delta_i} = \boldsymbol{x_i} - \boldsymbol{x_{best}}. \tag{6.42}$$

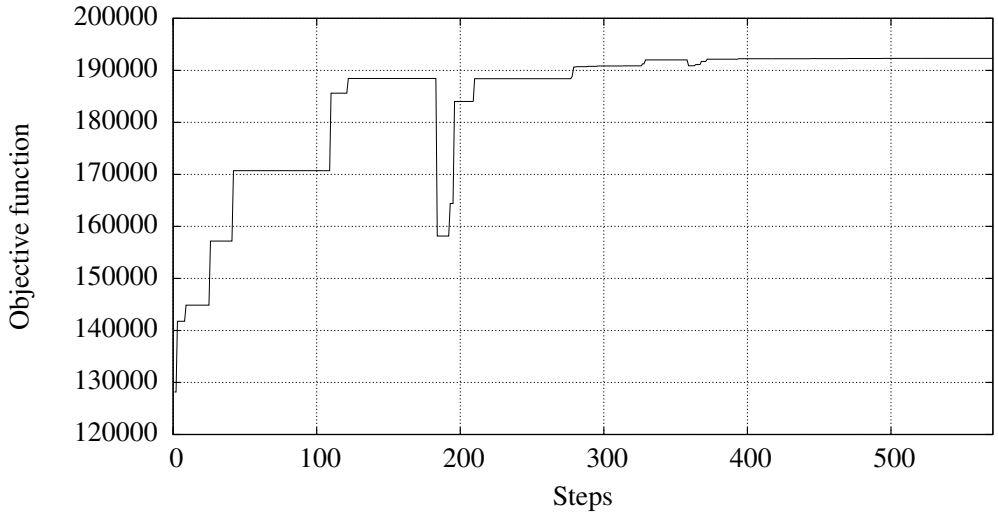


Figure 6.5: The convergence history for a typical optimization problem. The objective function value may drop when the accuracy of the FEM model is refined and the swarm is re-sampled.

6.2.5 Computational considerations

Using the finite element method coupled with a stochastic optimization algorithm may result in some computational difficulty if the models are not sufficiently small or if the algorithm does not converge rapidly enough. In order to ease this problem, it was decided to monitor the convergence of the swarm and break the optimization problem up into three distinct phases, based on the average particle velocity.

For the initial stage, while the swarm is still searching through the design space in a seemingly uncoordinated manner, it was deemed sufficient to use a rather coarsely-meshed finite element model which can be rapidly analyzed. For the second stage, when the swarm starts to converge on some optimum, the model was refined to a more acceptable accuracy. During the final stage of the analysis, the swarm maps out the space around a single optimum in great detail. This was undertaken with finely meshed models in order to ensure maximum accuracy.

A typical convergence history for one of the design problems that was solved in this study is shown in Figure 6.5. The Python implementation of the optimization algorithm is given in Appendix C.

Of course, the probability of finding the global optimum may be decreased by dividing the analysis up into stages, but if a global optimization strategy is implemented together with the PSOA, this effect is probably not of great consequence.

6.3 Global optimization strategy

For multi-modal problems, there is no guarantee that an optimization algorithm will find the global optimal solution. For this reason, it is difficult for a designer to report an optimal solution with much certainty unless some global optimization strategy is implemented. A multi-start strategy can be employed in this regards, to allow the optimization algorithm to find a series of optimal solutions from different starting positions in the design space.

In this study, a multi-start strategy was adopted together with a Bayesian stopping criterion that makes use of Bayes' theory of probability. This allows us as designers to assign a level of trust to our results, based on the outcomes from a series of starts. The program flow is shown in Figure 6.6.

In broad terms, the criterion calculates the probability of a given optimum being the global optimum by comparing the number of times that a solution has converged to the optimum in question vs. the number of times that the analysis has been repeated. This is done by making some mild assumptions regarding the relative sizes of the regions within the design space from where an analysis would converge to a particular solution. The explanation given below is taken from [32], which is largely based on the work of Snyman and Fatti [33].

6.3.1 Bayesian stopping criterion

Multi-start methods require a termination rule for deciding when to end the sampling, and to then use the current overall minimum function value $\tilde{f}$ as the approximation to the global minimum f^*, i.e.

$$\tilde{f} = \min \left\{ \hat{f}^j,\ j = 1, 2, \cdots \right\}, \tag{6.43}$$

where j represents the number of starting points to date, and where the $\hat{f}^j$ are assumed to be feasible local minima, $j = 1, 2, \cdots$.

Define the *region of convergence* of a local minimum $\hat{\mathbf{x}}$ as the set of all points $\mathbf{x}$ which, when used as starting points for a given algorithm, result in converge to $\hat{\mathbf{x}}$. Let R_k denote the region of convergence of local minimum $\hat{\mathbf{x}}^k$ and let α_k be the associated probability that a sample point be selected in R_k. The region of convergence and the associated probability for the global minimum $\mathbf{x}^*$ are denoted by R^* and α^* respectively. The following basic assumption – which is probably true for many combinations of different algorithms and many functions of practical interest – is now made:

$$\alpha^* \geq \alpha_k \text{ for all local minima } \hat{\mathbf{x}}^k. \tag{6.44}$$

The following theorem may then be proved [33]:

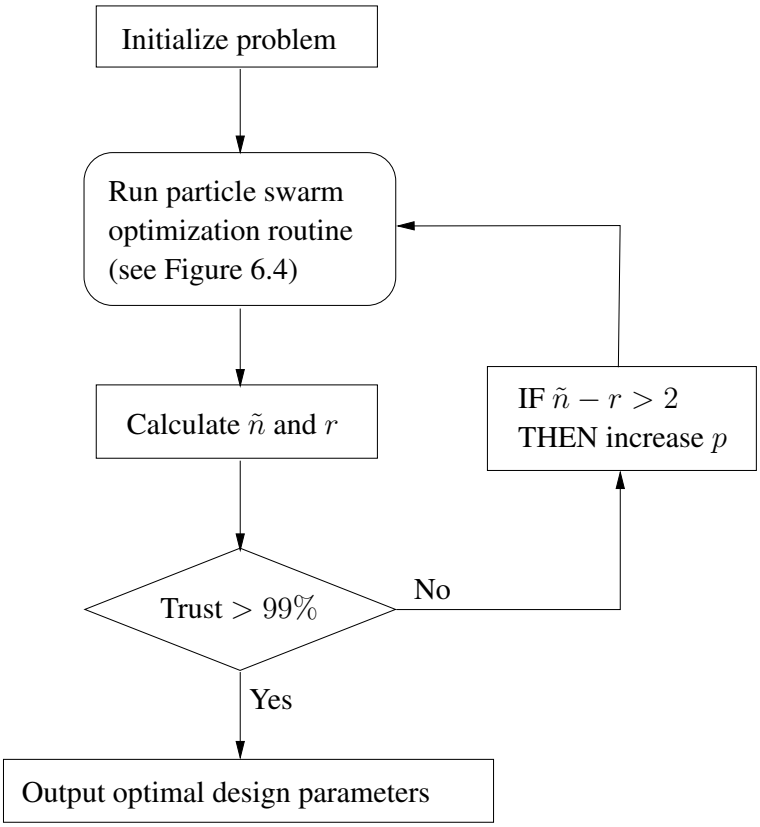


Figure 6.6: Overview of program flow for global optimization strategy.

Theorem 6.3.1 *Let r be the number of sample points falling within the region of convergence of the current overall minimum $\tilde{f}$ after $\tilde{n}$ points have been sampled. Then, under assumption* (6.44) *and a statistically non-informative prior distribution, the probability that $\tilde{f}$ corresponds to f^* may be obtained from*

$$\mathcal{P}_r\left[\tilde{f} = f^*\right] \geq q(\tilde{n}, r) = 1 - \frac{(\tilde{n}+1)!(2\tilde{n}-r)!}{(2\tilde{n}+1)!(\tilde{n}-r)!}, \tag{6.45}$$

where $\mathcal{P}_r$ is short for 'probability that'.

On the basis of Theorem 6.3.1, the adopted *stopping rule* becomes

$$\text{STOP when } \mathcal{P}_r\left[\tilde{f} = f^*\right] \geq q^*, \tag{6.46}$$

where q^* is some prescribed desired confidence level, typically chosen as 0.99 through 0.999. for the purposes of this study, a value of 0.99 was deemed sufficient.

The proof of Theorem (6.3.1) is given in Appendix D.

6.3.2 Swarm size and robustness

The majority of problems that were considered in this study could be solved with relatively few iterations of the particle swarm optimization routine. In general, a global optimization problem can be solved to a confidence level of 99% with only four iterations if the optimization routine converges on the same result for each iteration. That is, if $\tilde{n} = r = 4$, then

$$q(\tilde{n}, r) > 0.99 \tag{6.47}$$

according to Equation (6.45).

However, for a few problems the optimization routine failed to converge reliably on the same result. In such cases the particle swarm would typically converge prematurely on some local optimum which does not correspond to the global optimum solution to the problem. In order to mitigate this type of behavior, the size of the particle swarm was increased dynamically as soon as the difference between the number of evaluations $\tilde{n}$ and the number of evaluations that successfully converged r grew too large. This approach ensures a better sampling of the design space before the swarm starts to converge. The rule was implemented as

$$\begin{aligned} &\text{IF} \quad \tilde{n} - r > 2 \\ &\text{THEN} \quad p = 1.2 \times p, \end{aligned} \tag{6.48}$$

where again, p denotes the number of particles within a swarm. This rule was found to significantly increase the robustness of the program when confronted with some of the more difficult design problems.

Test problem	Function evaluations (avg.)	$r/\tilde{n}$	$\tilde{f} - f^*$	f^*
P1, Griewank 2	2326	4/4	2.02 e-10	0.0
P2, Griewank 10	5914	5/7	8.03 e-07	0.0
P3, Rosenbrock	3293	4/4	3.20 e-07	0.0
P4, Rastrigin	2421	4/4	2.37 e-09	-2.0

Table 6.1: Results from particle swarm optimization routine and global optimization strategy to a set of test problems. Here $\tilde{f}$ denotes the calculated solution and f^* denotes the exact solution. The four test problems are described in Appendix E.

6.3.3 Typical results

In order to demonstrate the efficiency of the Bayesian stopping criterion together with the particle swarm optimization algorithm, some example problems were solved. These problems form part of the extended Dixon-Szegö test set [28]. The test set is described in Appendix E and the results are summarized in Table 6.1. A swarm size of 5 particles was used for all of the problems.

To solve these analytical problems, the *particleSwarm.py* script given in Appendix C was adapted slightly to account for the different bounds on the design variables. Also, the 0.05% tolerance mentioned in Section 6.2.4 was removed, as the analytical functions do not generate any significant amount of numerical noise.

The performance of the PSOA can be highly dependant on the details of the implementation, and so the results above are not intended to reflect the general ability of the PSOA. Rather, it is intended to document the results obtained with this specific implementation, which was created only for the type of design problems encountered in this study.

Chapter 7

Conclusion

For the electricity grid of the future to accommodate useful levels of renewable energy, the problem of energy storage will first have to be solved. This is because renewable sources of energy are intermittent in nature, and storage is needed to link supply to demand.

Amongst the different energy storage technologies that are available, flywheels present one of the most energy efficient solutions. In addition to this, it is also environmentally friendly and does not suffer from the geographical limitations often associated with some of the other technologies. However, at present, the cost of flywheel energy storage will have to come down before the technology can be adopted on a wider scale. One way of achieving this effectively, is to increase the energy density of flywheel rotors.

In order to achieve high energy density in a flywheel rotor, it is possible to make use of several concentric layers of material within a single rotor. The material distribution within the rotor affects the stress distribution under centrifugal loading. In this way, the materials may be used to achieve a rotor design that is close to fully stressed at the point of failure.

In this study, the use of a geometric shape factor for cylindrical, composite rotors was reintroduced. This helps by giving a measure of how close a rotor is to its theoretical maximum energy density, and it allows us to strike a meaningful comparison between rotor designs even when they are not based on the same material set.

Thereafter a critical analysis was made of the design methods that are used in the available literature. It was found that several of the optimal rotors that have been reported in literature did not achieve a very high shape factor, and therefore it was concluded that it may be possible to improve on their energy densities. After scrutinizing the design methods, it was found that careful formulation of the design optimization problem is of utmost importance.

A novel formulation of the design optimization problem was proposed and its use was demonstrated on three example problems. By making use of this formulation, significant improvements were shown to be possible in certain cases.

With regards to the problem formulation, the main recommendations that came from this study are that volumetric energy density should be used as objective function, rather than weight energy density or the total stored energy. The design variables should include a relative overall thickness, and the individual layer thicknesses should also be defined in a relative sense. This allows for easy scaling of the solution. Lastly, the problem should be constrained in such a way that all candidate rotors are evaluated at the point of failure, and so it may be unwise to prescribe a fixed rotational velocity.

The design problems that were encountered in literature typically only consider very small sets of materials, probably due to the increased computational cost associated with an increase in the number of design variables. With the methods that are currently available, it is very difficult to find the best material from a large set, because this would necessitate costly repetition. For this reason, Chapter 5 discussed a novel proposition that serves to significantly reduce the cost of analysing large material sets for the type of problems that we are interested in. The use of the proposition was demonstrated by applying it to find the overall optimal rotor from a set of ten materials.

The shape factor of the optimal rotor was found to be relatively close to the theoretical maximum. This suggests that further improvements, which may be brought about by more expensive methods, such as shrink fitting or in-situ curing, may not be necessary if the material distribution is carefully designed according to the recommendations of this thesis.

In summary, the findings of this study may make it possible to design flywheel rotors with higher energy density than are available at present, whilst also simplifying the manufacturing processes that are necessary to obtain such high energy density designs. It is hoped that this study may ultimately contribute towards the development of flywheels that are cheap and effective enough to help integrate renewables onto the electricity grid.

Bibliography

[1] M. Krack, M. Secanell, and P. Mertiny, "Cost optimization of hybrid composite flywheel rotors for energy storage," *Structural & Multidisciplinary Optimization*, vol. 41, p. 779–795, 2010.

[2] A. C. Arvin and C. E. Bakis, "Optimal design of press-fitted filament wound composite flywheel rotors," *Composite Structures*, vol. 72, p. 47–57, 2006.

[3] S. K. Ha, D.-J. Kim, and T.-H. Sung, "Optimum design of multi-ring composite flywheel rotor using a modified generalized plane strain assumption," *International Journal of Mechanical Sciences*, vol. 43, pp. 993–1007, 2001.

[4] R. Baxter, *Energy Storage: A Nontechnical guide*. Oklahoma: Pennwell Corporation, 2006.

[5] S. van der Linden, "Bulk energy storage potential in the USA, current developments and future prospects," *Energy*, vol. 31, pp. 3446–3457, 2006.

[6] S. M. Schoenung and W. V. Hassenzahl, "Long- vs. short-term energy storage technologies analysis, a life-cycle cost study," tech. rep., Sandia National Laboratories, 2003. SAND2003-2783.

[7] P. R. S. Moorey, *Ancient Mesopotamian Materials and Industries: The Archaeological Evidence*. Oxford University Press, 1994.

[8] G. Genta, "Some considerations on the constant stress disc profile," *Meccanica*, vol. 24, pp. 235–248, 1989.

[9] G. Genta, *Kinetic Energy Storage*. London: Butterworth & Co. Ltd., 1985.

[10] G. Genta, "Spin tests on medium energy density flywheels," *Composites*, pp. 38–46, 1982.

[11] G. Genta, "The shape factor of composite material filament-wound flywheels," *Composites*, pp. 129–134, 1981.

[12] G. Genta, "On the design of thick rim composite material filament wound flywheels," *Composites*, vol. 15, pp. 49–55, 1984.

[13] E. L. Danfelt, S. A. Hewes, and T.-W. Chou, "Optimization of composite flywheel design," *International Journal of Mechanical Science*, vol. 19, pp. 69–78, 1977.

[14] G. G. Portnov, C. E. Bakis, and R. P. Emerson, "Some aspects of designing multirim composite flywheels," *Mechanics of Composite Materials*, vol. 40, pp. 397–408, 2004.

[15] B. C. Fabien, "The influence of failure criteria on the design optimization of stacked-ply composite flywheels," *Structural and Multidisciplinary Optimization*, vol. 33, pp. 507–517, 2007.

[16] C. T. Herakovich and Y. M. Tarnopol'skii, eds., *Handbook of Composites: Vol. 2 - Structures and Design*, ch. 10. Elsevier Science Publishers, 1989.

[17] S. K. Ha, J. H. Kim, and Y. H. Han, "Design of a hybrid composite flywheel multi-rim rotor system using geometric scaling factors," *Journal of Composite Materials*, vol. 42, pp. 771–785, 2008.

[18] J. L. Pérez-Aparicio and L. Ripoll, "Exact, integrated and complete solutions for composite flywheels," *Composite Structures*, vol. 93, p. 1404–1415, 2011.

[19] S. K. Ha, H. H. Han, and Y. H. Han, "Design and manufacture of a composite flywheel press-fit multi-rim rotor," *Journal of Reinforced Plastics and Composites*, vol. 27, pp. 953–965, 2008.

[20] P. J. Janse van Rensburg, D. W. Wood, and A. A. Groenwold, "Energy storage in composite flywheel rotors," in *2nd African Conference on Computational Mechanics* (D. Reddy, P. Nithiarasu, and A. Malan, eds.), January 2011.

[21] S. K. Ha, H. M. Jeong, and Y. S. Cho, "Optimum design of thick-walled composite rings for an energy storage system," *Journal of Composite Materials*, vol. 32, pp. 851–873, 1998.

[22] A. A. Groenwold and R. T. Haftka, "Optimization with non-homogeneous failure criteria like Tsai–Wu for composite laminates," *Structural Multidisciplinary Optimization*, vol. 32, pp. 183–190, 2006.

[23] S. K. Ha, H.-I. Yang, and D.-J. Kim, "Optimal design of a hybrid composite flywheel with a permanent magnet rotor," *Journal of Composite Materials*, vol. 33, pp. 1544–1575, 1999.

[24] O. C. Zienkiewicz, R. L. Taylor, and J. Z. Zhu, *The Finite Element Method: Its Basis and Fundamentals*. Butterworth-Heinemann, 6th ed., 2005.

[25] T. R. Chandrupatla and A. D. Belegundu, *Introduction to finite elements in engineering*. Prentice Hall, 2nd ed., 1997.

[26] R. D. Cook, *Finite element modelling for stress analysis*. John Wiley & Sons, 1995.

[27] S. W. Tsai and E. M. Wu, "A general theory of strength for anisotropic materials," *Journal of Composite Materials*, vol. 5, pp. 58–80, 1971.

[28] D. W. Wood, "A discourse concerning certain stochastic optimization algorithms and their application to the imaging of cataclysmic variable stars," Master's thesis, University of Pretoria, 2004.

[29] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proceedings of the 1995 IEEE International Conference on Neural Networks*, pp. 1942–1948, 1995.

[30] D. N. Wilke, S. Kok, and A. A. Groenwold, "Comparison of linear and classical velocity update rules in particle swarm optimization: notes on diversity," *International Journal for Numerical Methods in Engineering*, vol. 70, pp. 962–984, 2007.

[31] D. N. Wilke, S. Kok, and A. A. Groenwold, "Comparison of linear and classical velocity update rules in particle swarm optimization: notes on scale and frame invariance," *International Journal for Numerical Methods in Engineering*, vol. 70, p. 985–1008, 2007.

[32] A. A. Groenwold and L. F. P. Etman, *The 'Not-So-Short' manual for the SAOi algorithm*, version 0.7.13 ed., November 2010.

[33] J. A. Snyman and L. P. Fatti, "A multi-start global minimization algorithm with dynamic search trajectories," *Journal of Optimization theory and applications*, vol. 54, p. 121–141, 1987.

[34] R. Zielinsky, "A statistical estimate of the structure of multiextremal problems," *Mathematical Programming*, vol. 21, p. 348–356, 1981.

Appendix A

Stress distribution in example rotors

The stress distribution calculated for the three resulting rotors from Section 4.3 are given here for reference. The plots include initial thermal loading where applicable.

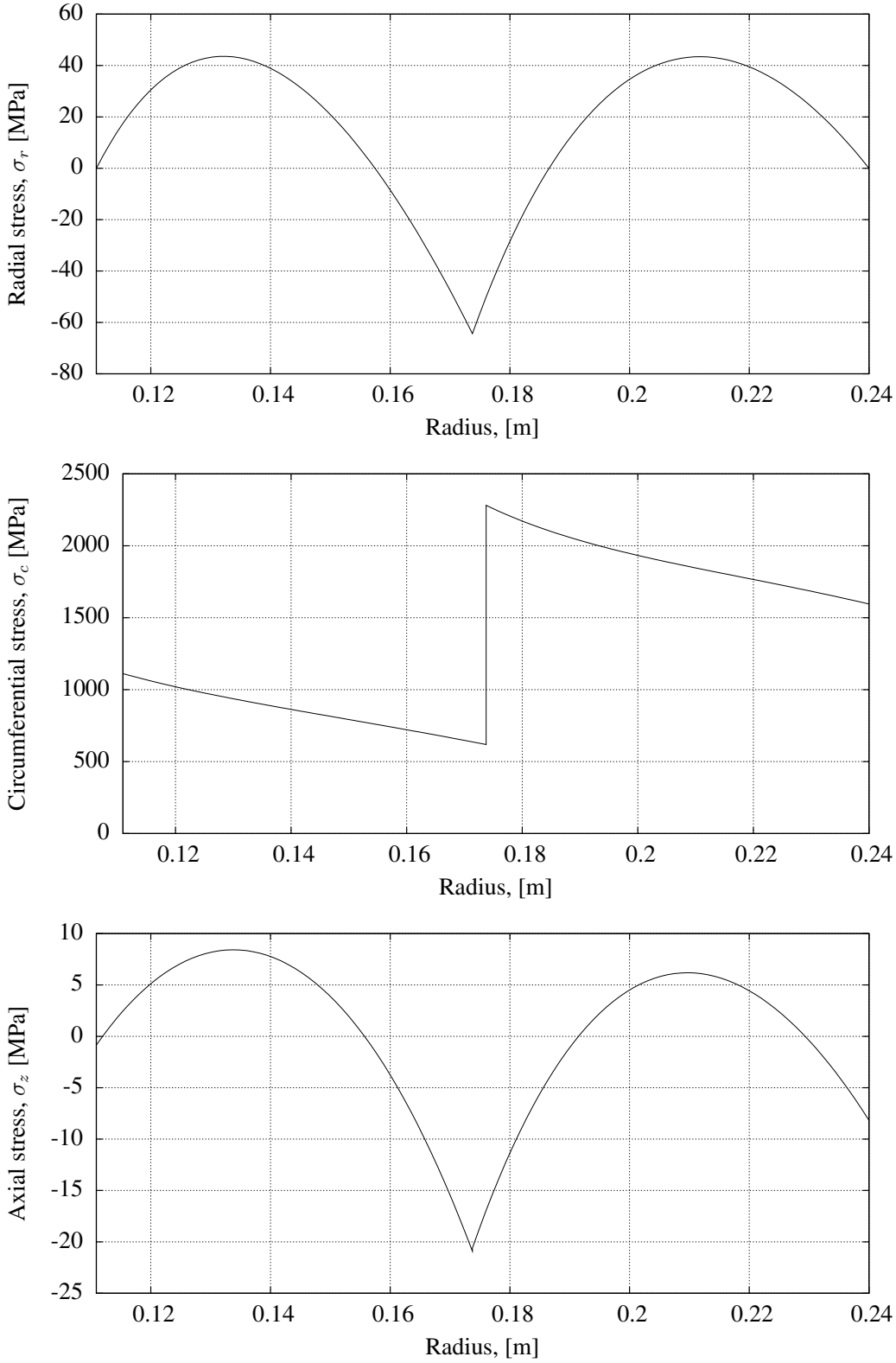


Figure A.1: The stress distribution through the mid-plane of the rotor named *Example 1* in Table 4.7, as given by axisymmetric FEM script.

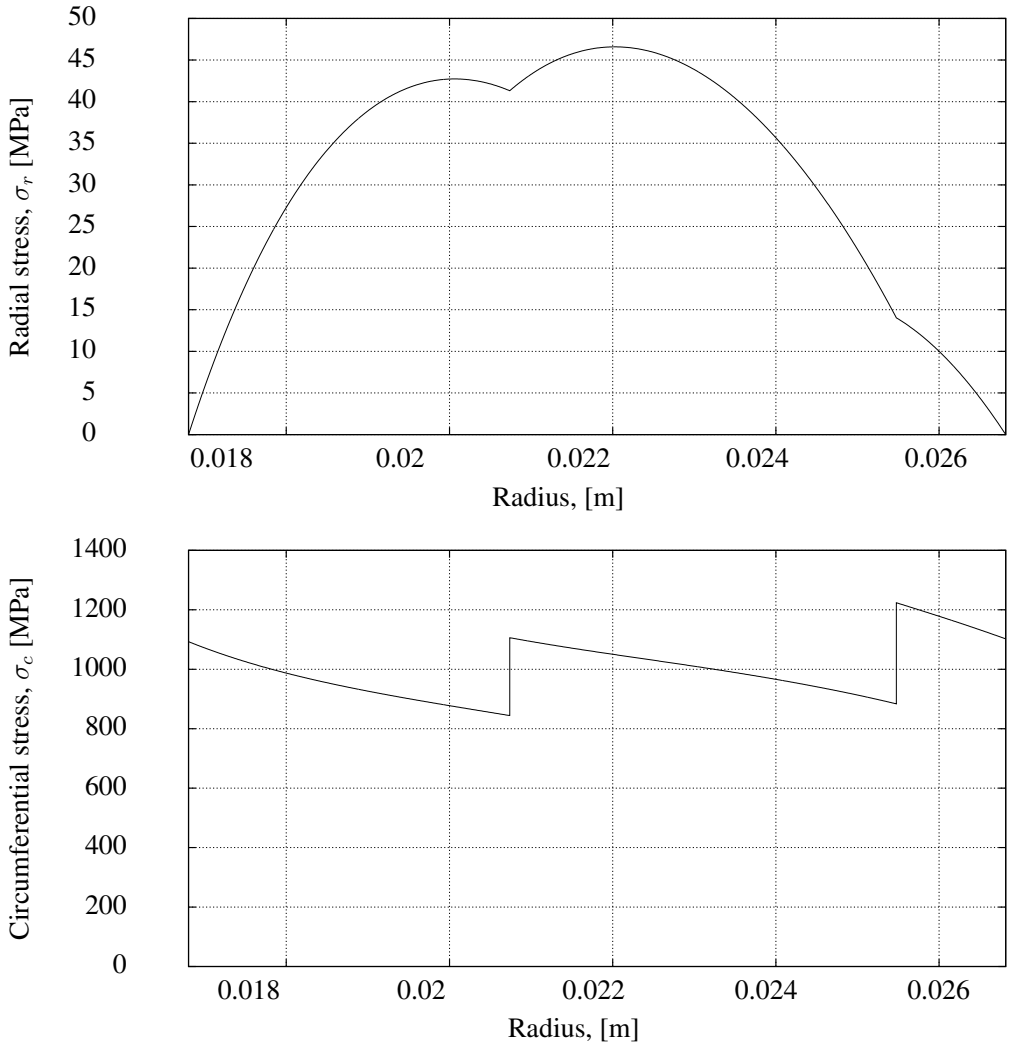


Figure A.2: The stress distribution through the rotor named *Example 2* in Table 4.7, as given by axisymmetric FEM script.

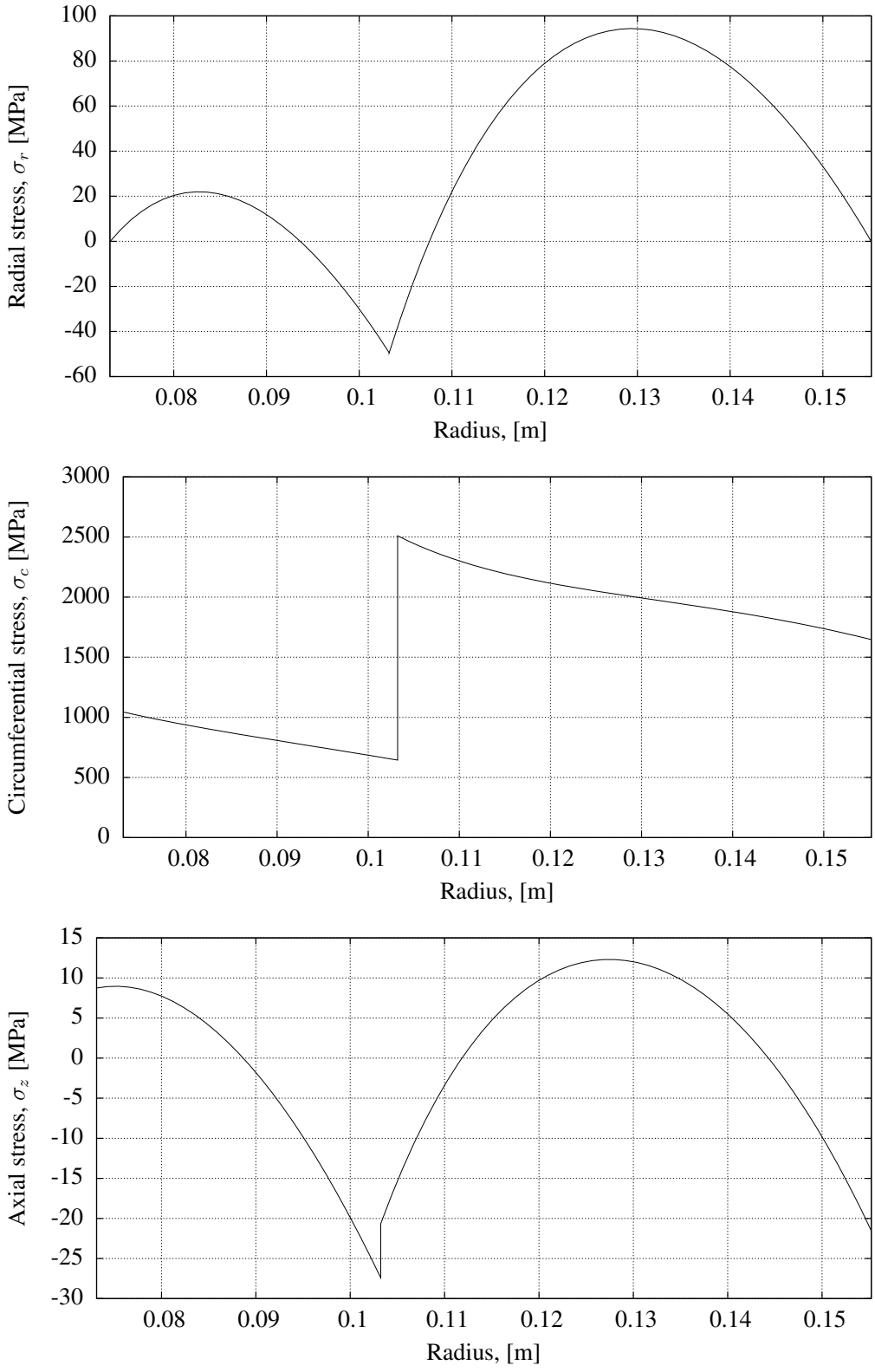


Figure A.3: The stress distribution through the mid-plane of the rotor named *Example 3* in Table 4.7, as given by axisymmetric FEM script.

Appendix B

Validation of axisymmetric FEM routine

B.1 Axisymmetric patch test

The patch test is used to verify whether a finite element solution will converge in the limit of mesh refinement. The basic principle of the test is to show that an element can accurately represent a state of constant stress under the correct loading. If this is the case, then it will be able to accurately represent any arbitrary stress state in the limit of mesh refinement. This is because the elements may be chosen to be sufficiently small for the stress field to be close enough to constant for the desired accuracy to be achieved. In the limit of mesh refinement, the solution is no longer just accurate, but exact.

The patch test, in it's various forms, is widely accepted as both necessary and sufficient as proof of convergence.

B.1.1 Stability

The first step is to demonstrate the stability of an element. This is done by calculating the eigenvalues and eigenvectors of the element stiffness matrix, before any constraints are imposed on the element. The four-noded, bilinear, axisymmetric element that was used for this study, is shown in Figure B.1. In constructing the model, the coordinates of the four points were taken to be $(1, 0)$, $(2, 0)$, $(2, 1)$ and $(1, 1)$ respectively. The following material properties were used: $E = 1 \times 10^6, v_{12} = 0.3$ and $\rho = 1.0$.

The eigenvalues of the element stiffness matrix are shown in Table B.1. It can be seen from the table that there is only one zero-valued eigenvalue, which means that only one rigid-body motion is captured by the model. This corresponds to axial

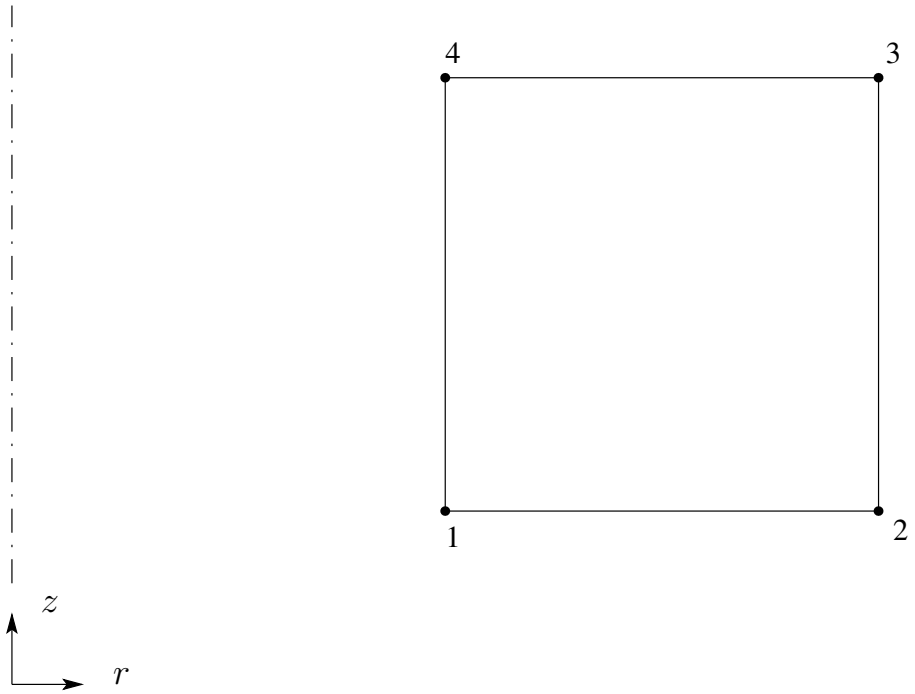


Figure B.1: A single four-noded element, used for analysing stability.

1	1.8948e+07	5	1.0894e+06
2	7.5011e+06	6	4.1198e+05
3	4.4677e+06	7	4.7123e+00
4	4.0168e+06	8	3.1824e-10

Table B.1: The eight eigenvalues of the stiffness matrix of a single four-noded bilinear element.

translation, which does not result in any strain within the element. The only other rigid-body motion associated with cylindrical bodies is rotation in the circumferential direction, however, this mode of movement is not captured by an axisymmetric model. From these results it can be concluded that the element that was used for this study is stable.

B.1.2 Consistency

To test whether an axisymmetric element can correctly model a state of constant stress, and whether it can do so consistently over element boundaries, a patch of elements, with at least one internal node needs to be be created. This patch can then be subjected to uniform pressure at its inside- and outside boundaries, while being restrained from axial movement on the top- and bottom boundaries. This configu-

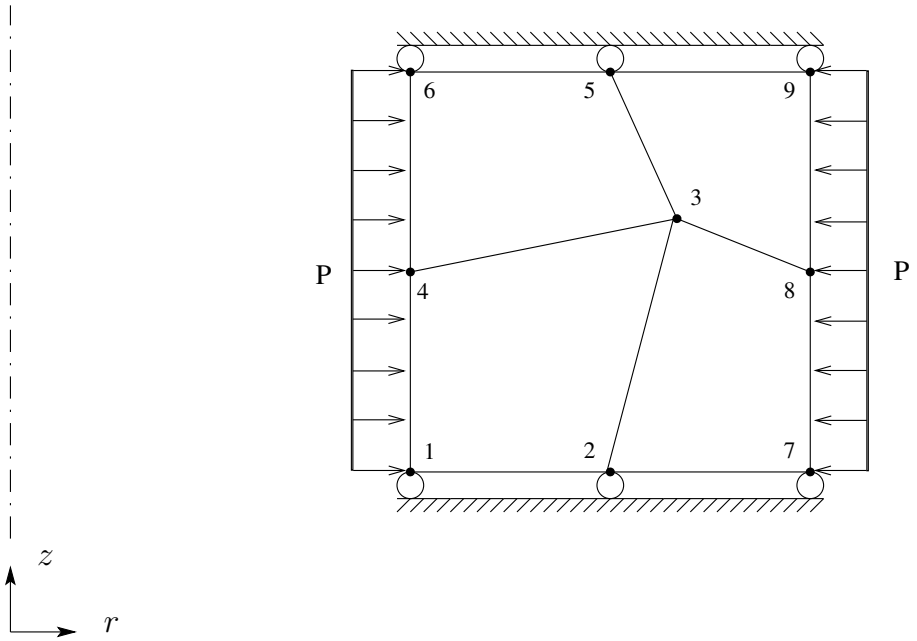


Figure B.2: A patch of four elements under constant pressure loading. This is used for verifying consistency.

ration should lead to a state of constant stress for all three principal directions, and it should induce no shear stress.

An example of such a patch is shown in Figure B.2, where the following values were used for the respective nodal coordinates: $(1.0, 0)$, $(1.5, 0)$, $(1.7, 0.7)$, $(1.0, 0.5)$, $(1.5, 1.0)$, $(1.0, 1.0)$, $(2.0, 0)$, $(2.0, 0.5)$ and $(2.0, 1.0)$.

The material properties from the previous section were used again, and the patch was loaded and constrained appropriately. The resulting stress is reported in Table B.2 for twelve points that were selected at random. It can be seen that the element that was used for this study is consistent, because it can represent a state of constant stress exactly (to within machine accuracy).

The four-noded bilinear axisymmetric element that was used for this study has been shown to be both stable and consistent, and therefore it satisfies the requirements of the patch test.

Coords (r, z)	σ_r	σ_c	σ_z	τ_{rz}
(1.63, 0.57)	-100000.00000000002	-99999.999999999898	-59999.999999999964	3.75e-16
(1.40, 0.52)	-100000.00000000005	-99999.999999999898	-59999.999999999956	4.35e-16
(1.56, 0.50)	-100000.00000000001	-99999.999999999913	-59999.999999999956	3.83e-16
(1.40, 0.98)	-100000.00000000002	-99999.999999999767	-59999.999999999985	9.54e-17
(1.34, 0.82)	-99999.999999999942	-99999.999999999753	-59999.999999999964	1.35e-16
(1.33, 0.99)	-100000.00000000002	-99999.999999999738	-59999.999999999971	1.15e-16
(1.85, 0.54)	-99999.999999999898	-99999.999999999869	-59999.999999999927	3.66e-16
(1.95, 0.30)	-99999.999999999913	-99999.999999999971	-59999.999999999985	4.02e-16
(1.86, 0.38)	-99999.999999999942	-99999.999999999956	-59999.999999999971	3.97e-16
(1.91, 0.80)	-100000.00000000000	-99999.999999999854	-59999.999999999949	1.60e-16
(1.86, 0.88)	-99999.999999999913	-99999.999999999782	-59999.999999999905	1.63e-16
(1.98, 0.79)	-99999.999999999942	-99999.999999999854	-59999.999999999927	1.68e-16
Exact solution	-100000	-100000	-60000	0.0

Table B.2: The stress state at twelve randomly selected points within the four-element patch.

B.2 Validation of results to example problems

In Section 4.3, the results are given for three example problems that were solved with the axisymmetric FEM routine that was written for this study. The details of the resulting optimal rotors are given in Table 4.7. Here all of the rotors are evaluated at the point of failure, where the safety factor given by the relevant failure criterion has a value of unity.

The stress distributions through each of these rotors are shown in Figures A.1 to A.3, as predicted by the axisymmetric FEM routine that was written for this study. In order to validate these results, the three optimal rotors were also constructed using the MD Nastran and Patran FEM package from MSC Software. The details of the Nastran/Patran FEM models are given below, after which a comparison is made between the results given by the two different FEM programs.

Geometry

Axisymmetric Tria6 elements were used for all three of the models. The first example made use of 100 elements along the radial direction, divided up into 46 for the inner layer and 54 for the outer layer, while 50 elements were used in the axial direction. For the second example problem the rotor was assumed to be in a state of plane strain, so the FEM model was constructed to be very thin in the axial direction (0.167 mm) with only a single element. However, for the radial direction the inner layer made use of 96 elements, while the middle- and outer layers had 112 and 32 elements respectively. The third example rotor made use of 80 elements along

the radial direction; 30 for the inner layer and 50 for the outer layer. In the axial direction the rotor used 120 elements.

Loading

The models were loaded by prescribing a rotational velocity and an initial temperature difference where applicable. The rotational velocity had to be converted to units of revolutions per second, for which the values for the three example problems were calculated to be 796.55, 5774.17 and 1341.9 respectively.

Constraints

Only one rigid-body mode had to be constrained for each model. This was done by prescribing zero axial displacement for the bottom, inside node of each model.

Results

The stress results from the Nastran/Patran FEM models were analyzed, and the same failure criteria were applied to them as was done for the other models. The safety factor at all of the nodes were calculated, and the minimum safety factor for each rotor was found to be 0.996 for the first example, 0.999 for the second, and 0.994 for the third.

The results show a relative difference of less than one percent between the results reported by the Python program and those calculated by the Nastran/Patran package.

B.3 Correlation to analytical solution

As an undergraduate, part of my final year's project was the derivation of analytical equations for describing the stress distribution in an axially thin ring, with circumferentially wound fiber composite material. It is an interesting exercise to see how well the predictions of the analytical equations correlate with those from the FEM code that was written for this study.

According to my derivations, if we let $n = \frac{r_i}{r_o}$, $R = \sqrt{r_i^2 + r_o^2}$ and $\lambda = \frac{E_1}{E_2}$, then the radial stress in a cylindrical composite rotor can be described by

$$\sigma_r = \frac{3 + \nu_{12}}{9 - \lambda} \rho \omega^2 \left(a - b - r^2 \right), \tag{B.1}$$

and the circumferential stress by

$$\sigma_c = \frac{3 + \nu_{12}}{9 - \lambda} \rho \omega^2 \left(\sqrt{\lambda} a + \sqrt{\lambda} b - \frac{\lambda + 3\nu_{12}}{3 + \nu_{12}} r^2 \right). \tag{B.2}$$

For the purpose of the two equations above, a is defined as

$$a = \left(\frac{1 - n^{3+\sqrt{\lambda}}}{1 - n^{2\sqrt{\lambda}}} \right) \left(1 + n^2 \right)^{-\frac{1}{2}\left(3 - \sqrt{\lambda}\right)} R^{3-\sqrt{\lambda}} r^{\sqrt{\lambda}-1}, \tag{B.3}$$

and b is defined as

$$b = \left(\frac{n^{2\sqrt{\lambda}} - n^{3+\sqrt{\lambda}}}{1 - n^{2\sqrt{\lambda}}} \right) \left[\left(1 + n^2 \right)^{-\frac{1}{2}\left(3 + \sqrt{\lambda}\right)} \right] R^{3+\sqrt{\lambda}} r^{-\sqrt{\lambda}-1}. \tag{B.4}$$

To test how well the results from the analytical solution above correlates with the axisymmetric FEM script that was used for this study, a plane-stress rotor was defined with $r_i = 0.2$ and $r_o = 0.5$. The T1000 material from Table 4.3 was used, and the model was set to spin at 20000 revolutions per minute with no temperature loading. A total of 50 elements were used in the radial direction, and only one element in the axial direction.

The stress predicted by both the analytical- and the FEM model is shown in Figure B.3. It can be seen that the results really do match very well. For the radial stress the maximum deviation was found to be 0.2 MPa and for the circumferential stress it was 1.4 MPa.

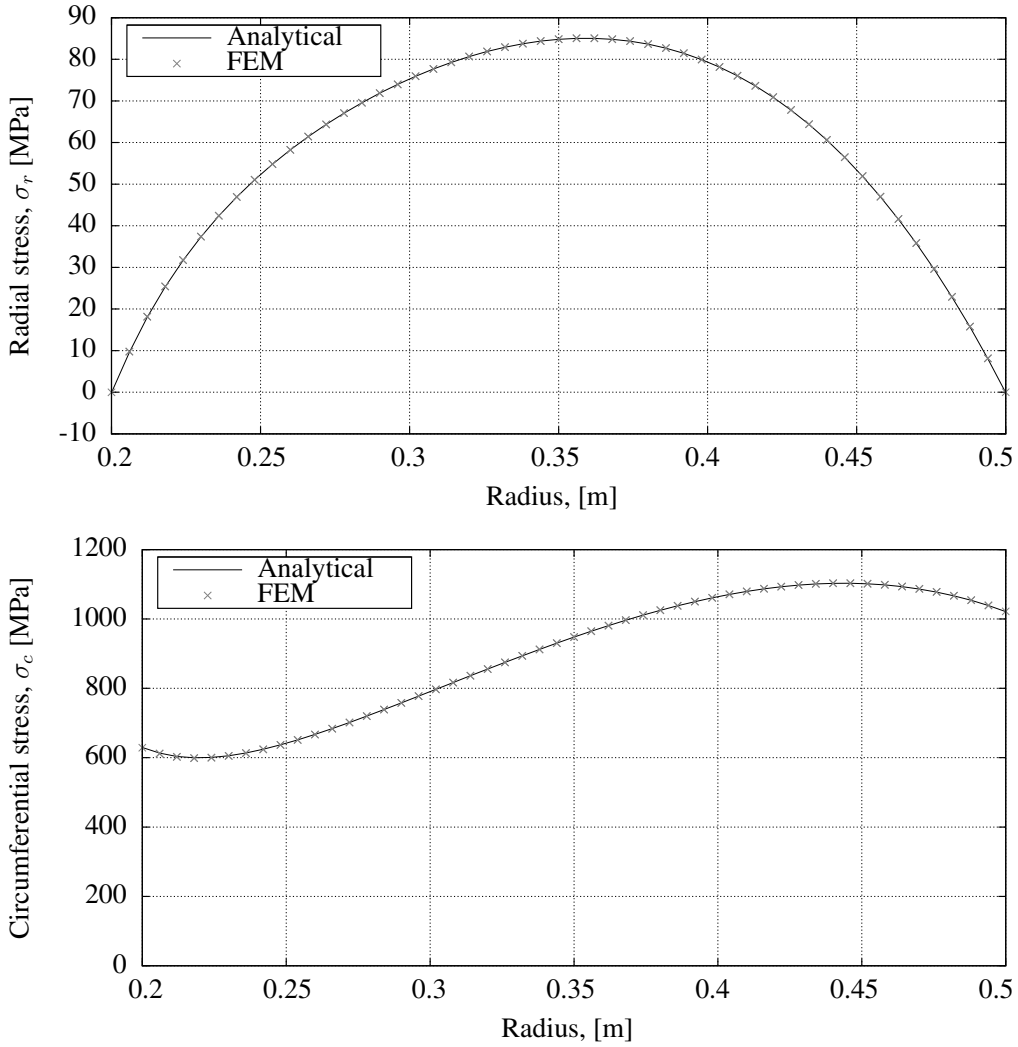


Figure B.3: The stress distribution through a thick single-material rotor.

Appendix C

Python code

Some extracts from the Python code that was written by the author are given here. Of particular interest is the code that implements an axisymmetric, four-noded, bi-linear element, and also the code that implements the particle swarm optimization algorithm. For the latter, the *parallel python* module was used to compute function evaluations in parallel.

Python version 2.6.6 was used throughout this study.

C.1 particleSwarm.py

```
import numpy as np
import functionCall
import sys, pp
import shelve
import materials as m
from datetime import datetime

start_time = datetime.now()
omega = 1.0
c1 = 2.0
c2 = 2.0
alpha = 0.99
beta = 0.98
h = 10
maxVelocity = 1.0
nr = 30

class Particle:
    def __init__(self, numDims):
        '''Initialise particle instance with random position and
           random velocity.'''

        # set random position
        self.position = np.zeros(numDims)
        self.setRandomPosition()

        # set random velocity
        self.velocity = np.zeros(numDims)
        self.setRandomVelocity()

        self.fitness = 'null'
        self.best = (self.position, 'null')
        return

    def setRandomPosition(self):
        '''Populate a random position vector, whilst ensuring that
           the position is feasible.'''

        # 1st design variable between zero and one
        self.position[0] = np.random.random()

        if len(self.position > 1):
            # layer thicknesses to a total of one
            sample = np.random.random(len(self.position))
            sample = sample/np.sum(sample)
            self.position[1::] = sample[1::]
        return

    def setRandomVelocity(self):
        '''Populate random velocity vector.'''

        self.velocity[0] = maxVelocity*(np.random.random() - 0.5)/0.5
```

```
        if (len(self.velocity) > 1):
            dims = len(self.velocity)-1
            sample = np.random.random(dims) - 0.5*np.ones(dims)
            sample = maxVelocity*sample/np.sum(np.abs(sample))
            self.velocity[1::] = sample
        return

    def isFeasible(self, position):
        '''Return True if a position falls within the design space,
            otherwise, return False.'''

        for i in range(len(position)):
            if position[i] < 0.0 or position[i] > 1.0:
                return False
        if (len(position) > 1):
            if np.sum(position[1::]) > 1.0:
                return False
        return True

    def velocityCheck(self):
        '''Scale down the velocity vector if it becomes too large.'''

        if np.sqrt(np.dot(self.velocity, self.velocity)) > maxVelocity:
            self.velocity *=
              maxVelocity/np.sqrt(np.dot(self.velocity, self.velocity))
        return

    def updatePosition(self):
        '''Implement position update rule.'''

        newPosition = self.position + self.velocity
        self.position = newPosition
        return

    def updateVelocity(self, globalBest, numDims):
        '''Implement velocity update rule.'''

        r1 = np.zeros((numDims, numDims))
        r2 = np.zeros((numDims, numDims))
        for i in range(numDims):
            r1[i][i] = np.random.random()
            r2[i][i] = np.random.random()
        pi = self.best[0]
        pg = globalBest[0]
        x = self.position
        self.velocity =
          omega*self.velocity + c1*np.dot(r1,(pi-x)) + c2*np.dot(r2,(pg-x))
        self.velocityCheck()
        return

class Swarm:
    def __init__(self, numParticles, numDims):
        '''Initialise swarm instance with the number of particles, and the
           dimensionality of the design problem as input.'''
```

```
        # initialise set of particles
        self.particles = []
        for i in range(numParticles):
            self.particles.append(Particle(numDims))

        self.best = (np.empty, 'null')
        self.worst = 'null'
        self.history = []
        return

    def evaluateFitness(self):
        '''Evaluate the fitness of each particle in the swarm. This can be
           done in parrallel for a multi-core machine.'''

        ppservers = ()
        job_server = pp.Server(ppservers=ppservers)
        jobs = []

        # evaluate list of feasible particles

        # PARALLEL IMPLEMENTATION FOR MULTI-CORE MACHINES
        for i in range(len(self.particles)):
            if self.particles[i].isFeasible(self.particles[i].position):
                jobs.append(
                  job_server.submit(
                    functionCall.objective,
                    (self.particles[i].position, nr),
                    (),
                    ("numpy", "structure", "interpolateRotor",
                      "evaluateMaxStress", "evaluateTsaiWu",
                      "optimization", "setParams", "shelve")
                  )
                )
            else:
                self.particles[i].fitness = -1.0
        j = 0
        for i in range(len(self.particles)):
            if self.particles[i].isFeasible(self.particles[i].position):
                self.particles[i].fitness = jobs[j]()
                j += 1
                # determine particle's best position to date
                if self.particles[i].best[1] == 'null':
                    self.particles[i].best =
                      (self.particles[i].position, self.particles[i].fitness)
                elif self.particles[i].fitness < self.particles[i].best[1]:
                    self.particles[i].best =
                      (self.particles[i].position, self.particles[i].fitness)
        job_server.destroy()

##        # SERIAL IMPLEMENTATION FOR SINGLE-CORE MACHINES
##        for part in self.particles:
##            part.fitness = functionCall.objective(part.position)
##            # determine particle's best position to date
##            if part.best[1] == 'null':
##                part.best = (part.position, part.fitness)
```

```
##            elif part.fitness < part.best[1]:
##                part.best = (part.position, part.fitness)

        # determine swarm's worst position at the current step
        self.worst = self.particles[0].fitness
        for part in self.particles:
            if part.fitness > self.worst:
                self.worst = part.fitness

        # determine swarm's best position to date
        for part in self.particles:
            if self.best[1] == 'null':
                self.best = (part.position, part.fitness)
            elif part.fitness < self.best[1]:
                self.best = (part.position, part.fitness)
        self.history.append(self.best)
        return

    def updatePosition(self):
        '''Call the position update rule for each particle in the swarm.'''

        for part in self.particles:
            part.updatePosition()
        return

    def updateVelocity(self):
        '''Call the velocity update rule for each particle in the swarm.'''

        numDims = len(self.particles[0].position)
        for i in self.particles:
            i.updateVelocity(self.best, numDims)
        return

    def timeStep(self):
        '''Execute one pseudo-timestep.'''

        self.updateVelocity()
        self.updatePosition()
        self.evaluateFitness()
        return

    def reduceInertiaVel(self):
        '''Reduce the value of the inertia and maximum-velocity parameters.'''

        global omega
        global maxVelocity
        omega *= alpha
        maxVelocity *= beta
        return

    def converge(self, margin = 0.01):
        '''Check whether all position vectors lie within a specified margin
           of the swarm's best position vector.'''

        out = True
```

```
        pos = self.best[0]
        for i in self.particles:
            diff = pos - i.position
            # 2-norm of the difference between positions
            distance = np.sqrt(np.dot(diff, diff))
            if distance > margin:
                out = False
                break
        return out

def main(growth = 1):
    '''Run one iteration of the PSOA algorithm. The problem's details are
       read in from the 'db_in' database, and the size of the swarm is
       determined depending on the dimensionality of the problem. However,
       the 'growth' parameter may be used to enlarge the swarm if the
       stopping criterion takes long to converge.'''

    # Set initial values for global variables.
    global nr
    nr = 40
    global omega
    omega = 1.0
    global maxVelocity
    maxVelocity = 1.0

    # Read input database
    db_in = shelve.open('current_materials')
    numDims = len(db_in['mats'])
    db_in.close()

    # Set problem stage, to determine the accuracy
      needed from the FEM model
    initialStage = True
    midStage = False
    finalStage = False

    # Set swarm size depending on the problem's dimensionality
    swarmSize = 5*growth
    if numDims == 3:
        swarmSize = 10*growth
    if numDims == 4:
        swarmSize = 100*growth
    if numDims == 5:
        swarmSize = 1000*growth

    # Initialize swarm instance
    s = Swarm(int(swarmSize), numDims)
    s.evaluateFitness()

    # Print feedback on the initial state of the swarm
    print 'Swarm, step 0'
    avgVel = 0
    for part in s.particles:
        avgVel += np.sqrt(np.dot(part.velocity, part.velocity))
    avgVel /= len(s.particles)
```

```
print 'omega = ', omega, 'maxVelocity = ',
  maxVelocity, 'avgVelocity = ', avgVel
print 'best position: ', s.best[0]
print 'best fitness: ', s.best[1]
print 'worst fitness: ', s.worst, '\n'
swarmHistory.append(s.best[1])

count_converge = 0
count_inertia = 0
count_catch = 0
prev = s.best[1]
i = 1
latch = True
outcome = 'exhausted'

# Iterate through pseudo-timesteps until convergence is achieved
while i < 1500:

    # Check for convergence
    if finalStage and count_catch > 200:
        outcome = 'aborted'
        print "INSUFFICIENT PROGRESS, SEARCH ABORTED"
        print 'best ', s.best[1], ' worst ', s.worst
        break
    if count_converge >= 10 and s.converge(0.005):
        outcome = 'converged'
        print 'CONVERGENCE ACHIEVED: best ',
          s.best[1], ' worst ', s.worst
        break

    # Execute timestep
    s.timeStep()
    swarmHistory.append(s.best[1])

    if s.best[1] > 1.0005*prev:
        count_converge += 1
        count_inertia += 1
        if nr == 100:
            count_catch += 1
    else:
        count_converge = 0
        count_inertia = 0
        count_catch = 0
    prev = s.best[1]
    if count_inertia == h:
        s.reduceInertiaVel()
        count_inertia = 0

    # Determine average velocity of swarm
    avgVel = 0
    for part in s.particles:
        avgVel += np.sqrt(np.dot(part.velocity, part.velocity))
    avgVel /= len(s.particles)

    # Print feedback on the current state of the swarm,
```

```
        every ten timesteps
    if i % 10 == 0:
        print 'Swarm, step ', i
        print 'nr = ', nr
        print 'omega = ', np.round(omega, 3), 'maxVel = ',
          np.round(maxVelocity, 3), 'avgVel = ', np.round(avgVel, 3)
        print 'best position: ', np.round(s.best[0], 4)
        print 'best fitness: ', np.round(s.best[1] , 4)
        print 'worst fitness: ', np.round(s.worst, 4)
        print 'Time elapsed: ', datetime.now() - start_time,'\n'

    # Check whether swarm has converged sufficiently to warrant an
      increase in accuracy from the FEM model
    if (initialStage):
        if avgVel < 0.33:
            # increase model's degrees of freedom
            nr = 50
            s.best = (np.empty, 'null')
            s.evaluateFitness()
            count_converge = 0
            count_inertia = 0
            initialStage = False
            midStage = True
    if (midStage):
        if avgVel < 0.11:
            # increase model's degrees of freedom
            nr = 70
            s.best = (np.empty, 'null')
            s.evaluateFitness()
            count_converge = 0
            count_inertia = 0
            midStage = False
            finalStage = True
    if (finalStage and latch):
        if avgVel < 0.037:
            # increase model's degrees of freedom
            latch = False
            nr = 100
            s.best = (np.empty, 'null')
            s.evaluateFitness()
            count_converge = 0
            count_inertia = 0
    i += 1

# Return result to output database,
  and also to the stopping criterion
outputFile = 'swarmHist.dat'
f=open(outputFile, 'w')
f.write('# step, F*\n')
for i in range(len(swarmHistory)):
    f.write(str(i+1) + ' ' + str(swarmHistory[i]) + '\n')
f.write('\n')
f.close()

return s.best, i, outcome, nr
```

C.2 elementQ4.py

```
import numpy as np
from numpy import linalg

def elasticity(mat):
    ''' Constitutive law for given material'''

    v1 = mat.v1
    E1 = mat.E1
    E2 = mat.E2
    v2 = v1*(E2/E1)
    if v1 < v2:
        print 'Error: material constants are ill-defined'
    G = mat.G2

    Qinv = np.array([[1./E2, -v1/E2, -v1/E1, 0],
                     [-v1/E2, 1./E2, -v1/E1, 0],
                     [-v2/E2, -v2/E2, 1./E1, 0],
                     [0, 0, 0, 1./G]])

    Q = np.linalg.inv(Qinv)
    return Q

def strainDisp((xsi, eta), coords):
    ''' Strain-displacement relationship. For a given point in
        natural coordinates, return value of B-matrix, shape
        functions, radius and determinate of the Jacobian matrix.'''

    # Shape functions:
    N1 = 0.25*(1-xsi)*(1-eta)
    N2 = 0.25*(1+xsi)*(1-eta)
    N3 = 0.25*(1+xsi)*(1+eta)
    N4 = 0.25*(1-xsi)*(1+eta)

    N = np.array([N1, N2, N3, N4])

    # Cast input coordinates into usable format
    [(r0, z0), (r1, z1), (r2, z2), (r3, z3)] = coords
    crds = np.array([[r0, z0],
                     [r1, z1],
                     [r2, z2],
                     [r3, z3]])

    # Derivatives of shape functions
    [dN1xsi, dN2xsi, dN3xsi, dN4xsi] =
        0.25*np.array([-(1-eta), (1-eta), (1+eta), -(1+eta)])
    [dN1eta, dN2eta, dN3eta, dN4eta] =
        0.25*np.array([-(1-xsi), -(1+xsi), (1+xsi), (1-xsi)])

    Dn = np.array([[dN1xsi, dN2xsi, dN3xsi, dN4xsi],
                   [dN1eta, dN2eta, dN3eta, dN4eta]])

    # Jacobian:
```

```
    J = np.dot(Dn, crds)
    detJ = np.linalg.det(J)
    invJ = np.linalg.inv(J)

    # Coordinate transformation to relate derivatives:
    Kappa = np.zeros((5, 5))
    Kappa[0:2, 0:2] = invJ
    Kappa[2:4, 2:4] = invJ
    Kappa[4, 4] = 1.

    # Calculate value of radius
    rk = N1*r0 + N2*r1 + N3*r2 + N4*r3

    H = np.array([[1., 0, 0, 0, 0],
                  [0, 0, 0, 1., 0],
                  [0, 0, 0, 0, 1./rk],
                  [0, 1., 1., 0, 0]])

    Q = np.array([[dN1xsi, 0, dN2xsi, 0, dN3xsi, 0, dN4xsi, 0],
                  [dN1eta, 0, dN2eta, 0, dN3eta, 0, dN4eta, 0],
                  [0, dN1xsi, 0, dN2xsi, 0, dN3xsi, 0, dN4xsi],
                  [0, dN1eta, 0, dN2eta, 0, dN3eta, 0, dN4eta],
                  [N1, 0, N2, 0, N3, 0, N4, 0]])

    # Calculate value of strain-displacement matrix
    B = np.dot(np.dot(H, Kappa), Q)
    return B, N, rk, detJ

def stiffness(mat, coords):
    ''' Calculate element stiffness matrix.'''

    E = elasticity(mat)
    k = np.zeros((8, 8))
    # two-point Gauss integration
    for xsi in [-1/np.sqrt(3), 1/np.sqrt(3)]:
        for eta in [-1/np.sqrt(3), 1/np.sqrt(3)]:
            [B, N, rk, detJ] = strainDisp((xsi, eta), coords)
            k += 2*np.pi*np.dot(np.dot(B.T, E), B)*rk*detJ
    return k

def bodyForce(mat, coords, omega):
    ''' Calculate element force vector caused by centrifugal loading.'''

    ff = np.zeros(8)
    E = elasticity(mat)
    # two-point Gauss integration
    for xsi in [-1/np.sqrt(3), 1/np.sqrt(3)]:
        for eta in [-1/np.sqrt(3), 1/np.sqrt(3)]:
            [B, N, rk, detJ] = strainDisp((xsi, eta), coords)
            NN = np.array([[N[0], 0, N[1], 0, N[2], 0, N[3], 0],
                           [0, N[0], 0, N[1], 0, N[2], 0, N[3]]])
            b = np.array([rk*mat.rho*omega**2, 0])
            ff += 2*np.pi*np.dot(b, NN)*rk*detJ
    return ff
```

```
def initialForce(mat, coords, delT):
    ''' Calculate initial force vector due to temperature difference.'''

    ff = np.zeros(8)
    E = elasticity(mat)
    Eps_init = delT*np.array([mat.alpha2, mat.alpha2, mat.alpha1, 0])
    # two-point Gauss integration
    for xsi in [-1/np.sqrt(3), 1/np.sqrt(3)]:
        for eta in [-1/np.sqrt(3), 1/np.sqrt(3)]:
            [B, N, rk, detJ] = strainDisp((xsi, eta), coords)
            ff += 2*np.pi*np.dot(np.dot(B.T, E), Eps_init.T)*rk*detJ
    return ff


def stress(E, displacement, coords, (xsi, eta)):
    ''' Calculate stress vector at point in element.'''

    [B, N, rk, detJ] = strainDisp((xsi, eta), coords)
    sigma = np.dot(E, np.dot(B, displacement))
    return sigma

def initStress(E, displacement, coords, (xsi, eta), Eps1, Eps2, delT):
    ''' Calculate stress vector caused by initial temperature difference.'''

    Eps_0 = delT*np.array([Eps2, Eps2, Eps1, 0])
    [B, N, rk, detJ] = strainDisp((xsi, eta), coords)
    sigma_0 = np.dot(E, np.dot(B, displacement)) - np.dot(E, Eps_0)
    return sigma_0
```

Appendix D

Proof of the Bayesian stopping criterion

The outline of the proof of Theorem (6.3.1) which is presented here is taken from [32], which follows closely the presentation given by Snyman and Fatti [33]. It can be shown that this proof is a generalization of the procedure proposed by Zielinski [34].

Given $\tilde{n}^*$ and α^*, the probability that at least one point, $\tilde{n} \geq 1$, has converged to f^* is

$$\mathcal{P}_r[\tilde{n}^* \geq 1|\tilde{n}, r] = 1 - (1-\alpha^*)^{\tilde{n}} \; . \tag{D.1}$$

In the Bayesian approach, we characterize the uncertainty about the value of α^* by specifying a prior probability distribution for it. This distribution is modified using the sample information (namely, $\tilde{n}$ and r) to form a posterior probability distribution. Let $p_*(\alpha^*|\tilde{n}, r)$ be the posterior probability distribution of α^*. Then,

$$\begin{aligned} \mathcal{P}_r[\tilde{n}^* \geq 1|\tilde{n}, r] &= \int_0^1 \left(1 - (1-\alpha^*)^{\tilde{n}}\right) p_*(\alpha^*|\tilde{n}, r) d\alpha^* \\ &= 1 - \int_0^1 (1-\alpha^*)^{\tilde{n}} p_*(\alpha^*|\tilde{n}, r) d\alpha^* . \end{aligned} \tag{D.2}$$

Now, although the r sample points converge to the current overall minimum, we do not know whether this minimum corresponds to the global minimum of f^*. Utilizing (6.44), and noting that $(1-\alpha)^{\tilde{n}}$ is a decreasing function of α, the replacement of α^* in the above integral by α yields

$$\mathcal{P}_r[\tilde{n}^* \geq 1|\tilde{n}, r] \geq \int_0^1 \left(1 - (1-\alpha)^{\tilde{n}}\right) p(\alpha|\tilde{n}, r) d\alpha \; . \tag{D.3}$$

Now, using Bayes theorem, we obtain

$$p(\alpha|\tilde{n}, r) = \frac{p(r|\alpha, \tilde{n}) p(\alpha)}{\displaystyle\int_0^1 p(r|\alpha, \tilde{n}) p(\alpha) d\alpha} \; . \tag{D.4}$$

Since the $\tilde{n}$ points are sampled at random and each point has a probability α of converging to the current overall minimum, r has a binomial distribution with parameters α and $\tilde{n}$. Therefore

$$p(r|\alpha,\tilde{n}) = \binom{\tilde{n}}{r}\alpha^r(1-\alpha)^{\tilde{n}-r} \; . \tag{D.5}$$

Substituting (D.5) and (D.4) into (D.3) gives:

$$\mathcal{P}_r[\tilde{n}^* \geq 1|\tilde{n},r] \geq 1 - \frac{\displaystyle\int_0^1 \alpha^r(1-\alpha)^{2\tilde{n}-r}p(\alpha)d\alpha}{\displaystyle\int_0^1 \alpha^r(1-\alpha)^{\tilde{n}-r}p(\alpha)d\alpha} \; . \tag{D.6}$$

A suitable flexible prior distribution $p(\alpha)$ for α is the beta distribution with parameters a and b. Hence,

$$p(\alpha) = (1/\boldsymbol{\beta}(a,b))\,\alpha^{a-1}(1-\alpha)^{b-1}, \qquad 0 \leq \alpha \leq 1. \tag{D.7}$$

Using this prior distribution gives

$$\begin{aligned}\mathcal{P}_r[\tilde{n}^* \geq 1|\tilde{n},r] \;&\geq\; 1 - \frac{\Gamma(\tilde{n}+a+b)\;\Gamma(2\tilde{n}-r+b)}{\Gamma(2\tilde{n}+a+b)\;\Gamma(\tilde{n}-r+b)} \\ &=\; 1 - \frac{(\tilde{n}+a+b-1)!\;(2\tilde{n}-r+b-1)!}{(2\tilde{n}+a+b-1)!\;(\tilde{n}-r+b-1)!}.\end{aligned}$$

Assuming a prior expectation of 1, (viz. $a = b = 1$), we obtain

$$\mathcal{P}_r[\tilde{n}^* \geq 1|\tilde{n},r] \;=\; 1 - \frac{(\tilde{n}+1)!\;(2\tilde{n}-r)!}{(2\tilde{n}+1)!\;(\tilde{n}-r)!},$$

which is the required result.

Appendix E

Analytical test problems

The following four analytical problems form part of the extended Dixon-Szegö test set [28].

P1, Griewank 2

Objective function

$$f(\boldsymbol{x}) = \sum_{i=1}^{2} x_i^2/200 - \prod_{i=1}^{2}[\cos(x_i/\sqrt{i}) + 1] \tag{E.1}$$

Search domain

$$D = \{(x_1, x_2) \in \mathbb{R}^2 : -100.0 \leq x_i \leq 100.0, \;\; i = 1, 2\} \tag{E.2}$$

Solution

$$\boldsymbol{x} = (0.0, 0.0), \;\; f^* = 0.0 \tag{E.3}$$

P2, Griewank 10

Objective function

$$f(\boldsymbol{x}) = \sum_{i=1}^{10} x_i^2/4000 - \prod_{i=1}^{10}[\cos(x_i/\sqrt{i}) + 1] \tag{E.4}$$

Search domain

$$D = \{(x_1, x_2, \cdots, x_{10}) \in \mathbb{R}^{10} : -600.0 \leq x_i \leq 600.0, \;\; i = 1, 2, \cdots, 10\} \quad \text{(E.5)}$$

Solution

$$\boldsymbol{x} = (0.0, \cdots, 0.0), \;\; f^* = 0.0 \quad \text{(E.6)}$$

P3, Rosenbrock

Objective function

$$f(\boldsymbol{x}) = 100(x_2 - x_1^2)^2 + (x_1 - 1)^2 \quad \text{(E.7)}$$

Search domain

$$D = \{(x_1, x_2) \in \mathbb{R}^2 : -5.0 \leq x_i \leq 5.0, \;\; i = 1, 2\} \quad \text{(E.8)}$$

Solution

$$\boldsymbol{x} = (1.0, 1.0), \;\; f^* = 0.0 \quad \text{(E.9)}$$

P4, Rastrigin

Objective function

$$f(\boldsymbol{x}) = x_1^2 + x_2^2 - \cos(18x_1) - \cos(18x_2) \quad \text{(E.10)}$$

Search domain

$$D = \{(x_1, x_2) \in \mathbb{R}^2 : -1.0 \leq x_i \leq 1.0, \;\; i = 1, 2\} \quad \text{(E.11)}$$

Solution

$$\boldsymbol{x} = (0.0, 0.0), \;\; f^* = -2.0 \quad \text{(E.12)}$$